

Федеральное государственное бюджетное учреждение науки  
Санкт-Петербургский институт информатики и автоматизации  
Российской академии наук  
(СПИИРАН)

На правах рукописи



**Жукова Наталия Александровна**

**МНОГОУРОВНЕВЫЙ СИНТЕЗ АВТОМАТНЫХ МОДЕЛЕЙ ОБЪЕКТОВ  
МОНИТОРИНГА**

Специальность: 05.13.01 - Системный анализ, управление и обработка  
информации (технические системы)

Диссертация на соискание ученой степени доктора  
технических наук

Научный консультант  
доктор технических наук, профессор,  
Осипов Василий Юрьевич

Санкт-Петербург–2019

## Содержание

<b>ВВЕДЕНИЕ .....</b>	<b>4</b>
<b>ГЛАВА 1. АНАЛИЗ ПРОЦЕССОВ ПОСТРОЕНИЯ МОДЕЛЕЙ ОБЪЕКТОВ МОНИТОРИНГА.....</b>	<b>10</b>
1.1. Цели, задачи и возможности построения моделей объектов мониторинга .....	10
1.2. Условия построения моделей объектов мониторинга .....	11
1.3. Анализ моделей объектов, процессов и программ мониторинга .....	15
1.4. Анализ моделей, методов и средств предварительной обработки данных мониторинга .....	18
1.5. Анализ методов синтеза моделей объектов, процессов и программ мониторинга.....	26
1.6. Формулировка проблемы и логическая схема исследования .....	29
1.7. Выводы .....	30
<b>ГЛАВА 2. ОСНОВЫ ТЕОРИИ МНОГОУРОВНЕВОГО СИНТЕЗА АВТОМАТНЫХ МОДЕЛЕЙ ОБЪЕКТОВ МОНИТОРИНГА.....</b>	<b>32</b>
2.1. Концепция многоуровневого синтеза автоматных моделей объектов мониторинга .....	32
2.2. Обобщенная формулировка задачи многоуровневого синтеза моделей объектов мониторинга ..	36
2.3. Система показателей и критериев эффективности многоуровневого синтеза моделей объектов мониторинга .....	36
2.4. Многоуровневые автоматные модели объектов мониторинга .....	39
2.5. Типовые формулировки задач многоуровневого синтеза моделей объектов мониторинга .....	45
2.6. Обобщенный алгоритм многоуровневого синтеза моделей объектов мониторинга .....	48
2.7. Выводы .....	50
<b>ГЛАВА 3. МЕТОДЫ МНОГОУРОВНЕВОГО СИНТЕЗА МОДЕЛЕЙ ОБЪЕКТОВ, ПРОЦЕССОВ И ПРОГРАММ МОНИТОРИНГА.....</b>	<b>52</b>
3.1. Методы многоуровневого синтеза автоматных моделей объектов мониторинга .....	52
3.2. Метод многоуровневого синтеза автоматных моделей процессов мониторинга.....	55
3.3. Методы многоуровневого синтеза автоматных моделей программ мониторинга.....	56
3.4. Методы и модели многоуровневой трансформации данных мониторинга .....	57
3.5. Методы и модели обеспечения трансформации данных мониторинга .....	60
3.6. Выводы .....	82
<b>ГЛАВА 4. МЕТОДЫ И МОДЕЛИ РАЗРАБОТКИ ПРОБЛЕМНО- И ПРЕДМЕТНО- ОРИЕНТИРОВАННЫХ СИСТЕМ ПОСТРОЕНИЯ МОДЕЛЕЙ ОБЪЕКТОВ ПО ДАННЫМ МОНИТОРИНГА.....</b>	<b>84</b>
4.1. Метод разработки проблемно- и предметно- ориентированных систем построения моделей объектов по данным мониторинга .....	84
4.2. Модели гибкой архитектуры ППСМ.....	87
4.3. Метод гибкого проектирования и сопровождения ППСМ .....	97
4.4. Архитектурные метрики систем ППСМ.....	106
4.5. Платформа ППСМ .....	111
4.6. Выводы по главе 4 .....	127

<b>ГЛАВА 5. МЕТОДИКИ И РЕЗУЛЬТАТЫ ПОСТРОЕНИЯ МОДЕЛЕЙ ОБЪЕКТОВ ПО ДАННЫМ МОНИТОРИНГА ДЛЯ РЕШЕНИЯ ПРИКЛАДНЫХ ЗАДАЧ.....</b>	<b>129</b>
5.1. Методики построения моделей объектов по данным мониторинга для решения прикладных задач .....	129
5.2. Средства для построения моделей объектов по данным мониторинга в прикладных предметных областях.....	131
5.3. Рекомендации по построению моделей объектов по данным мониторинга для решения прикладных задач .....	134
5.4. Результаты построения моделей объектов по данным мониторинга в прикладных предметных областях.....	140
5.5. Выводы .....	160
<b>ЗАКЛЮЧЕНИЕ.....</b>	<b>163</b>
<b>СПИСОК СОКРАЩЕНИЙ И УСЛОВНЫХ ОБОЗНАЧЕНИЙ .....</b>	<b>165</b>
<b>СПИСОК ЛИТЕРАТУРЫ.....</b>	<b>166</b>
<b>ПРИЛОЖЕНИЕ 1. РЕЗУЛЬТАТЫ ПОСТРОЕНИЯ МОДЕЛЕЙ ОБЪЕКТОВ ПО ДАННЫМ МОНИТОРИНГА НА ПРИМЕРЕ ОБЛАСТИ ТЕЛЕКОММУНИКАЦИЙ.....</b>	<b>210</b>
П.1. Постановка задачи построения моделей объектов по данным мониторинга в области телекоммуникаций.....	210
П.2. Исходные данные для построения моделей объектов по данным мониторинга в области телекоммуникаций.....	212
П.3. Результаты построения моделей объектов по данным мониторинга в области телекоммуникаций .....	215
П.4. Результаты построения моделей процессов мониторинга в области телекоммуникаций .....	230
П.5. Оценка эффективности построения моделей наблюдаемых объектов и процессов мониторинга в области телекоммуникаций .....	246
П.6. Рекомендации по построению моделей наблюдаемых объектов и процессов мониторинга в области телекоммуникаций .....	264
<b>ПРИЛОЖЕНИЕ 2. АКТЫ ВНЕДРЕНИЯ.....</b>	<b>269</b>
<b>ПРИЛОЖЕНИЕ 3. СВИДЕТЕЛЬСТВА О РЕГИСТРАЦИИ ПРОГРАММ.....</b>	<b>284</b>

## Введение

**Актуальность темы.** Объектами мониторинга могут являться технические, биологические, природные и другие объекты. В технической области средства мониторинга традиционно используются для наблюдения за специализированными объектами, например, ракетно-космическая, авиационная и другая техника. Моделирование этих объектов позволяет снизить вероятность ошибок в управлении, которые могут привести к необратимым негативным последствиям, срыву выполнения возлагаемых на объекты задач. Существенное внимание вопросам синтеза моделей объектов мониторинга также уделяется в области телекоммуникаций. За последние несколько лет телевизионные и другие сети вышли на новый уровень развития. Одновременно усложнились задачи мониторинга этих сетей и управления ими. За счет синтеза моделей телекоммуникационных сетей возможно обеспечить их надежную и безопасную работу. В области природных и биологических объектов синтез моделей объектов мониторинга может позволить избежать различных природных катастроф, гибели живых организмов.

Исходными данными для построения моделей объектов являются результаты измерений параметров объектов, а также данные о внешней среде. Объекты имеют многоуровневую структуру, изменяющуюся во времени. Их состояние и поведение, как правило, описываются в дискретном времени и дискретном пространстве состояний.

Формируемые в настоящее время модели объектов мониторинга (ОМ) во многом не удовлетворяют требованиям практики по оперативности и точности их построения. На их построение из-за сложности подлежащих решению задач синтеза затрачивается много людских, а также иных ресурсов. При этом получаемые модели не всегда полны и во многом не адекватны реальным объектам. С учетом этого, проблема синтеза моделей объектов мониторинга является актуальной.

Для выхода на качественно новый уровень построения моделей объектов мониторинга требуется наличие соответствующих методов их автоматического синтеза. С учетом многоуровневой изменяющейся структуры наблюдаемых объектов, их описания в дискретном времени и дискретном пространстве состояний на основе результатов измерений их параметров, необходим автоматический многоуровневый синтез перестраиваемых автоматных моделей ОМ.

**Степень разработанности темы.** В настоящее время известен ряд методов, которые в той или иной мере применимы для построения моделей объектов мониторинга. Среди традиционных решений следует отметить методы моделирования сложных систем (Л.А. Растрин, Л. Берталанфи, А.И. Уемов, М. Месарович, Д. Клир), методы оптимального и адаптивного управления (А.А. Фельдбаум, А.Я. Лернер, Л.И. Розоноэр, А.Г. Бутковский, С.В.



Емельянов, Я.З. Цыпкин, Б.Н. Петров, А.А. Красовский); методы на основе идей традиционного искусственного интеллекта, ситуационного управления, связывания и трансформации данных (Д.А. Поспелов, С. Рассел, П. Норвиг, Г. Пятецкий-Шапиро, Е. Блаш, А. Штейнберг, В.К. Финн, С.А. Айвазян, В.Н. Вапник, А.С. Мандель, С. Малла, Н.Г. Загоруйко). К последним разработкам относятся модели и методы интеллектуального мониторинга, оценки их качества (Р.М. Юсупов, Б.В. Соколов, М.Ю. Охтилев), когнитивного поведения и управления знаниями, поиска и обработки информации (В.И. Городецкий, А.В. Смирнов, В.В. Александров, А.Л. Тулупьев). Непосредственно к автоматическому синтезу автоматных моделей различных объектов относятся работы Д.А. Робинсона, С. Чанга, Р. Ли, С.Ю. Маслова, У.Э. Х. Тыгу, С.Н. Баранова, В.Ю. Осипова.

Несмотря на то, что в последние годы в области синтеза моделей объектов мониторинга получены значительные результаты, они не лишены серьезных ограничений. Среди них выступают высокие требования к входным данным, высокая сложность решаемых задач, необходимость значительного участия человека в процессе моделирования, не проработанность автоматического многоуровневого синтеза моделей объектов мониторинга.

Таким образом, можно утверждать, что *уровень развития теории синтеза во многом не удовлетворяет современным потребностям практики в построении моделей объектов мониторинга.*

Имеет место *актуальная научная проблема разработки основ теории и методов многоуровневого автоматического синтеза автоматных моделей объектов мониторинга, отвечающих современным потребностям практики по снижению вычислительной сложности этого синтеза.*

**Цель работы:** развитие теории многоуровневого автоматического синтеза моделей объектов мониторинга, кратно снижающей вычислительную сложность этого синтеза.

**Объект исследования:** системные связи, закономерности функционирования и развития объектов мониторинга.

**Предмет исследования:** методы и средства синтеза автоматных моделей объектов мониторинга.

Для достижения поставленной цели были решены следующие **научные задачи:**

1. Анализ предметной области построения моделей наблюдаемых объектов по данным мониторинга их состояний. Постановка научной проблемы, определение требований к синтезируемым моделям и методам их синтеза.

2. Разработка основ теории многоуровневого автоматического синтеза автоматных моделей объектов мониторинга.

3. Разработка методов многоуровневого автоматического синтеза автоматных моделей объектов мониторинга.

4. Разработка методов многоуровневого автоматического синтеза автоматных моделей процессов и программ мониторинга для построения моделей объектов.

5. Разработка методов и моделей многоуровневой адаптивной трансформации данных мониторинга для построения моделей объектов.

6. Разработка методов и моделей разработки проблемно- и предметно -ориентированных систем построения моделей объектов по данным мониторинга.

7. Разработка системы методик построения моделей объектов по данным мониторинга для решения прикладных задач.

8. Апробация моделей, методов и методик автоматического синтеза автоматных моделей объектов мониторинга.

**Для решения задач диссертационного исследования использованы методы** и модели синтеза моделей объектов, процессов и программ, построения сложных систем, системного анализа, математического моделирования, искусственного интеллекта, адаптивной обработки данных, математической статистики, инженерии знаний, проектирования, разработки и сопровождения информационных систем.

**Научная новизна.** Научная новизна полученных в диссертационной работе результатов состоит в следующем.

1. Разработаны основы новой теории многоуровневого автоматического синтеза автоматных моделей объектов мониторинга, отличающейся оригинальными: концептуальной моделью синтеза многоуровневых перестраиваемых автоматных моделей, системой показателей и критериев эффективности, предусматривающей оценку полноты моделей и сложности их построения, формализмами математического описания иерархических относительно-конечных автоматов, математическими формулировками задач многоуровневого синтеза.

2. Разработаны новые методы многоуровневого автоматического синтеза автоматных моделей объектов мониторинга, которые, в отличие от существующих, являются индуктивно - дедуктивными, базируются на новых постановках задач и методах их решения, включающих метод многоуровневого индуктивного синтеза, позволяющего строить модели объектов по поступающим от них данным, и метод многоуровневого дедуктивного синтеза, позволяющий доказывать существование новых, ранее не рассматривавшихся моделей, в пространстве, построенном в результате индуктивного синтеза.

3. Для построения моделей объектов предложены новые методы многоуровневого автоматического синтеза автоматных моделей процессов и программ мониторинга,

отличающиеся низкой вычислительной сложностью, обеспеченной за счет разработки оригинальных методов, позволяющих строить процессы мониторинга, основанные на доказательстве их существования с применением прямого нисходящего многоуровневого вывода, и программы мониторинга на основе обратного многоуровневого вывода.

4. Разработаны новые методы и модели многоуровневой трансформации данных, обеспечивающие возможность контентно адаптивной обработки результатов мониторинга за счет описания процессов обработки в общем виде и их поэтапной детализации до уровня программно-реализуемых с учетом содержания данных и условий синтеза моделей объектов.

5. Предложены оригинальные методы и модели разработки проблемно- и предметно-ориентированных систем, описываемых в виде иерархии согласованных архитектурных и онтологических моделей, что позволяет применять гибкие методологии при их проектировании, создании и сопровождении.

6. Разработана новая система программно реализуемых методик построения моделей объектов мониторинга, отличающаяся от существующих тем, что обеспечивает полноту синтезируемых прикладных моделей и низкую сложность их синтеза, позволяющая решать практические задачи с учетом их классов, показателей и критериев эффективности, используемых для оценки формируемых результатов, а также типов данных, обрабатываемых в предметных областях.

**Теоретическая и практическая ценность** работы заключается в том, что новый аппарат многоуровневого синтеза дает возможность автоматически строить модели наблюдаемых объектов по данным мониторинга, обеспечивающие успешное решение задач прогнозирования, управления и других в прикладных предметных областях. Для построения моделей объектов по данным мониторинга разработаны информационные, архитектурные и программные компоненты, позволяющие строить проблемно- и предметно-ориентированные программные системы. Разработаны и внедрены системы построения моделей объектов для нескольких предметных областей. Результаты их эксплуатации предметными специалистами показали эффективность применения новых систем на практике. Они позволили существенно расширить состав решаемых прикладных задач, сократить время, затрачиваемое на их решение. Также новые системы обеспечили повышение точности и достоверности результатов решения значительной части прикладных задач по сравнению с существующими подходами.

**Положения, выносимые на защиту:**

1. Основы теории многоуровневого автоматического синтеза автоматных моделей объектов мониторинга.

2. Методы многоуровневого автоматического синтеза автоматных моделей объектов мониторинга.

3. Методы многоуровневого автоматического синтеза автоматных моделей процессов и программ мониторинга для построения моделей объектов.

4. Методы и модели многоуровневой адаптивной трансформации данных мониторинга для построения моделей объектов.

5. Методы и модели разработки проблемно- и предметно- ориентированных систем построения моделей объектов по данным мониторинга.

6. Система методик построения моделей объектов по данным мониторинга для решения прикладных задач.

**Достоверность результатов исследования.** Достоверность основных теоретических результатов обеспечивается за счет того, что новый аппарат многоуровневого автоматического синтеза основан на известной теории относительно конечных операционных автоматов. Синтез многоуровневых моделей объектов мониторинга опирается на классические методы индуктивного и дедуктивного синтеза. Синтезируемые процессы и программы не приводят к искажению информации о наблюдаемых объектах за счет формального доказательства обоснованности этих процессов и программ на каждом из шагов синтеза. Кроме того, достоверность полученных научных результатов подтверждается результатами проведенного моделирования и результатами опытной эксплуатации внедренных систем в нескольких предметных областях.

**Внедрение результатов работы** Приведенные в диссертации результаты внедрены в научно-исследовательские и опытно-конструкторские работы, реализованные в НИЦ СПб ЭТУ, СПИИРАН и СПбГЭТУ “ЛЭТИ“ в период с 2006 по 2017 года, включая ОКР «Поиск-СНВ», ОКР «Адекватность», ОКР «Ленинградка», ОКР «Радиент», ОКР «Диагностика- НИЦ», ОКР «Математика-ПИК», ОКР «Модернизация», ОКР «Интеграция», ОКР «Перспектива-ПИК», ОКР «Моренос», ОКР «ИАС-М», ОКР «Признак», ОКР «Указчик-ВКО/ПТК ОДПП и ОРИ», НИР «Листва-2000-НЦ», НИР «Радиент», ОКР «Алеврит», ОКР «Автоматизм», «СППР «Автоматизм», ОКР « Галтель - Алеврит»», НИР ONR-Global# 62909-12-1-7013 «Decision Making Support System for Arctic Exploration, Monitoring and Governance», НИР«Эстафета-Ф-СПИИРАН» и другие. Кроме того, полученные результаты использованы при создании прототипов семантической медицинской информационной системы для ФГБУ «НМИЦ им. В. А. Алмазова» Минздрава России. Прототипы разработаны на базе лаборатории в Санкт-Петербургском национальном исследовательском университете информационных технологий, механики и оптики. Также новые модели и методы многоуровневого синтеза применялись при разработке программных продуктов в коммерческих организациях (ОАО Zodiac Interactive, VIA Technologies и др.) Общее число внедрений составляет 29, из них 10 НИР, 1 НИОКР, 17 ОКР.

Результаты диссертационного исследования также используются в учебном процессе в Университете ИТМО, Санкт-Петербургском государственном электротехническом университете «ЛЭТИ» им. В.И. Ульянова (Ленина) и в Институте медицинского образования Центра Алмазова.

Внедрения результатов диссертационной работы подтверждены актами.

**Апробация работы.** Результаты апробированы автором на 28 международных, национальных и региональных научных конференциях.

**Публикации.** По теме диссертации опубликовано 5 монографий, 36 статьи в журналах, рекомендованных ВАК при Минобрнауки России, 20 статей в изданиях, индексируемых в SCOPUS и Web of Science, 3 программы для ЭВМ, 3 учебных пособия.

**Структура и объем работы.** Диссертационная работа состоит из введения, пяти глав, заключения, списка использованной литературы и приложений. Общий объем работы составляет 286 страниц, в том числе 484 источника литературы.

Логика исследования предусматривала анализ текущего состояния моделей, методов и средств построения моделей объектов мониторинга в предметных областях и обобщение полученных результатов. Для обеспечения выявленных потребностей предусматривалась разработка основ теории многоуровневого синтеза моделей объектов и научно методологического аппарата, позволяющего синтезировать формальные модели. Для синтеза конкретных моделей предусматривалась разработка моделей и методов трансформации данных мониторинга. Разработанный аппарат синтеза положен в основу нового класса систем построения моделей объектов по данным мониторинга. Для применения предлагаемого аппарата синтеза на практике предусматривалась разработка системы методик построения моделей объектов по данным мониторинга, включающей общую методику и частные методики для конкретных предметных областей. Результаты эксплуатации разработанных и внедренных систем обрабатываются и определяют направления дальнейшего развития предлагаемого аппарата синтеза.

## Глава 1. Анализ процессов построения моделей объектов мониторинга

### 1.1. Цели, задачи и возможности построения моделей объектов мониторинга

Построение моделей объектов по данным мониторинга проводится в целях решения практических задач, требующих прогнозирования состояния наблюдаемых объектов и управления ими. Задачи такого моделирования в предметных областях формулируются применительно к техническим и природным объектам.

В настоящее время значительная часть задач моделирования объектов в предметных областях решается прикладными специалистами. Анализ их работы показал, что в основе процессов построения моделей лежат принципы системного подхода. При построении моделей объектов специалистами используются их когнитивные способности к которым относятся способности построения иерархий, связывания событий и явлений, построения моделей. Результирующие модели обладают следующими *свойствами*:

- модели объектов имеют многоуровневую структуру;
- между элементами в моделях присутствуют взаимосвязи;
- модели представляют объекты в соответствии с поставленными целями моделирования.

При построении моделей объектов мониторинга решается *пять классов задач*.

*Задача класса А. Обработка результатов наблюдений.* Определение преобразований над исходными данными о наблюдаемых объектах, а также над результатами их обработки.

*Задача класса В. Восстановление моделей объектов по имеющимся данным.* Восстановление структур объектов и их изменений по данным мониторинга.

*Задача класса В. Управление мониторингом.* Определение состава собираемых данных, частоты контроля параметров наблюдаемых объектов.

*Задача класса Г. Построение представлений моделей объектов.* Модели объектов представляются в форме, требуемой для решения задач практики. Представления могут формироваться для выявления отклонений в состоянии объектов, их локализации, определения причин возникновения. Они будут отличны от представлений, необходимых для заблаговременного выявления возможных отклонений.

*Задача класса Д. Исследование статистических данных мониторинга.* Построение статистических моделей объектов и среды для заданной предметной области.

Задачи классов А и Г относятся к задачам области обработки и анализа данных, задача класса В - к области мониторинга, классов Б и Д - к области построения моделей сложных объектов.

За последние годы в области построения моделей объектов активно ведутся прикладные исследования. Инициаторами исследований, как правило, выступают предметные специалисты. Имеющиеся у них средства моделирования часто не применимы для решения существующих

задач в их новых постановках, а также для решения новых, ранее не рассматривавшихся задач. Разрабатываемые новые решения в большинстве случаев носят локальный характер, они жестко ориентированы на конкретные задачи, часто требуют модернизации, что приводит к необоснованно большим затратам материальных ресурсов. Результаты детального анализа текущего состояния области построения моделей объектов мониторинга, проведенного по данным за десятилетний период, позволили установить следующее. Возможности такого построения крайне ограничены из-за несовершенства соответствующей теории и применяемых технологий. К настоящему времени сложилась ситуация, когда остро востребована разработка новых решений для области построения моделей наблюдаемых объектов по данным мониторинга.

## **1.2. Условия построения моделей объектов мониторинга**

Решение многих практических задач [1], [2] построения моделей объектов мониторинга показало, что объекты мониторинга имеют сложную структурную организацию. Часто такие объекты являются пространственно соотнесенными. В их состав входят десятки, сотни, а иногда и тысячи элементов. Все эти элементы некоторым образом связаны друг с другом и находятся в постоянном взаимодействии. Одни составляющие могут оказывать существенное влияние на другие. Большое число составляющих приводит к неоднородности объектов в целом. Отдельно наблюдаемые объекты могут являться элементами других, еще более сложных объектов, о которых требуется получать информацию. Объекты могут объединяться в локальные и глобальные сети. Значительное влияние на состояние и поведение объектов может оказывать окружающая среда.

Возможности по построению моделей объектов ограничиваются наличием исходных данных об этих объектах, внешней среде, моделей связанных с ними объектов; наличием ресурсов для сбора и обработки данных; допустимыми значениями точности и достоверности формируемых решений, допустимым временем решения задач.

Основным источником данных о наблюдаемых объектах и внешней среде являются результаты измерений их параметров. Проблемы получения данных и их использования отнесены к ряду важных проблем достаточно давно. За последние годы интерес к ним существенно увеличился. Проводимые в этой области исследования носят масштабный характер. В них вовлечены ведущие мировые компании. Ежегодно публикуются многие аналитические отчеты, дающие оценки текущего состояния области обработки и анализа данных, а также определяющие перспективы ее развития [3], [4]. Данные отчетов показывают, что возможности технических средств достаточны для производства данных. Однако, возможности потребления данных [5], несмотря на появление сотен новых консалтинговых компаний, создания десятков аналитических платформ, остаются на низком уровне [6]. В

результате, наблюдается тенденция к непрерывному росту объема получаемых данных и уменьшению доли обрабатываемых данных.

Сложность обработки во многом определяется свойствами данных. При обработке необходимо учитывать следующее:

1. Структурно-сложный характер потоков данных. Для сбора данных применяются разнообразные технические средства, предусматривающие как активный, так и пассивный сбор данных. Эти средства могут быть установлены непосредственно на объектах или находиться на значительных расстояниях. Для передачи данных формируются потоки. В результате объединения данных, собранных многими устройствами, получаемые потоки имеют сложную внутреннюю структуру, в ряде случаев трудно выявляемую. Как правило, потоки имеют большой объем и поступают в реальном времени.

2. Многомерность и связанность данных. Данные, получаемые с объектов, представляют собой результаты измерений значений параметров состояния элементов, входящих в состав объектов. Отдельные элементы могут характеризоваться многими параметрами. Между параметрами могут существовать связи, в том числе, неочевидные.

3. Отсутствие прямых данных об объектах. Часто можно получить только косвенную информацию о состоянии объектов через наблюдения за отдельными его параметрами (как индивидуальными, так и групповыми). По результатам измерений определяются характеристики объектов. Обработка и анализ характеристик позволяют оценить состояние объекта в целом.

4. Необходимость учета априорной информации о наблюдаемых объектах, получаемых из различных источников. При обработке, анализе и интерпретации результатов измерений необходимо учитывать весь объем доступной информации (экспертных и статистических данных), имеющей прямое или косвенное отношение к объекту.

5. Необходимость в применении интеллектуальных методов и средств обработки данных совместно с классическим статистическим аппаратом. Необходимо применение методов совместной обработки данных, включая методы восстановления возможных зависимостей.

6. Ограниченные возможности построения формальных моделей для описания данных и их обработки. Для построения моделей обработки необходимо прогнозировать поведение параметров наблюдаемых объектов. Однако сложность объектов не позволяет построить полное пространство возможных состояний параметров и переходов между ними.

При построении моделей объектов осуществляется обработка данных о состоянии и поведении объектов, передаваемых в информационных потоках, с учетом данных, определяющих условия наблюдений, сбора и обработки. Содержание информационных



потоков, поступающих от объектов, носит название *контента* информационных потоков. Условия обработки контента определяют *контекст* построения моделей объектов мониторинга.

Основные свойства, характеризующие контент, показаны на рисунке 1.1. Отдельные свойства пояснены ниже.

*Содержательность.* Контент может содержать данные об измерениях, данные об объектах, данные о ситуациях. Эти данные могут использоваться при обработке результатов измерений параметров.

*Носитель.* Носителями информации являются объекты, группы объектов, данные о которых содержатся в передаваемых потоках.

*Структурированность.* Контент может иметь четкую структуру или требовать дополнительных процедур по ее выявлению.

*Относительность.* Передаваемые данные могут иметь пространственные, временные или событийные привязки.



Рисунок 1.1 – Основные свойства контента

*Инвариантность.* Отражает изменение ценности передаваемых данных во времени. Ценность данных может оставаться постоянной, уменьшаться с течением времени и, соответственно, требовать пополнения, или быть условно-постоянной.

*Качество.* Качество информации определяется такими параметрами как число ошибок, длина интервалов, на которых данные потеряны и т.п. Информация может иметь оценку качества или таковая может отсутствовать.

*Релевантность.* Контент может представлять собой фактические данные, результаты их обработки и другие. Данные могут относиться к одному или нескольким объектам.

*Согласованность.* Результаты измерений различных параметров объектов, выполненные различными средствами, могут различаться.

*Актуальность.* Оценивается временной интервал от момента получения измерений до момента начала их обработки с учетом скорости устаревания данных.

*Формализованность.* Данные могут быть формализованными, частично формализованными и неформализованными. Примером формализованных данных являются бинарные потоки, сформированные в соответствии со стандартом IRIG 106 [7]. Такой поток содержит описание передаваемых данных в заголовочной части или сопровождается отдельным описанием. Другим примером являются данные по отдельным измерениям, передаваемые по фиксированным согласованным протоколам в текстовом или бинарном виде. К формализованным также относятся данные, представляемые в соответствии со стандартами XML/XML Schema, RDF/RDF Schema, OWL и т.д. [8].

*Интерпретируемость.* Различают: данные, сопровождаемые описаниями, интерпретация которых возможна на машинном уровне; данные, сопровождаемые описаниями, интерпретация которых не возможна или ограничена; данные, не содержащие описания, которое может быть использовано при машинной обработке.

*Полнота.* Отражает достаточность контента поступающих потоков для решения прикладных задач.

На свойства получаемой об объектах информации, а также на возможность ее получения могут оказывать влияние негативные факторы.

При обработке контента рассматривается контекст. Возможно рассмотрение нескольких контекстов (таблица. 1.1).

Таблица 1.1 – Контексты построения моделей объектов мониторинга

Контекст	Наименование	Краткое описание
<i>HC</i>	Контекст исторических данных	Статистические данные об исследуемых объектах
<i>GC</i>	Контекст геоинформационной среды	Информация о состоянии и особенностях внешней среды, включает пространственно-временные и другие параметры
<i>DC</i>	Контекст предметной области	Информация, относящаяся к предметной области. Это могут быть сведения об объектах, способах их использования и другие
<i>RC</i>	Контекст доступных алгоритмических, программных и технических средств обработки	Информация о средствах, которые могут быть использованы при решении поставленных задач. Такими средствами являются программные библиотеки алгоритмов, информационные системы, физические сервера и другие
<i>SC</i>	Специализированные контексты	Эти контексты определяются для конкретных предметных областей. Например, для области медицины таким контекстом является история болезни пациента

Подробный анализ предложенных контекстов приведен в [9].

Таким образом, условия построения моделей объектов определяются свойствами контента и контекста.

### **1.3. Анализ моделей объектов, процессов и программ мониторинга**

#### **1.3.1. Анализ моделей объектов мониторинга**

Традиционно модели объектов, включая модели объектов мониторинга, подразделяются на:

- модели, формализующие объекты в непрерывном времени и непрерывном пространстве состояний;
- модели, описывающие объекты в дискретном времени и непрерывном пространстве состояний;
- модели, отражающие объекты в непрерывном времени и дискретном пространстве состояний;
- модели, формализующие объекты в дискретном времени и дискретном пространстве состояний. Такие модели называют автоматными моделями.

Модели объектов мониторинга предлагается характеризовать параметрами, представленными на рисунке 1.2. Выбор параметров моделей осуществляется исходя из требований, которые определяются последующим использованием моделей.

Анализ классических моделей [10], [11], показал, что они применимы при мониторинге простых объектов. Возможности их использования при мониторинге сложных объектов ограничены из-за низкого уровня иерархичности, малого сжатия обрабатываемой информации и ряда других.

За последние годы разработан ряд новых моделей для решения задач мониторинга. Значительный интерес представляет новый подход к комплексной автоматизации процессов мониторинга и управления состояниями сложных технических объектов [12]. Подход основан на полимодельном многокритериальном описании процессов, концепциях и принципах теорий управления структурной динамикой, распознавании образов, недоопределенных вычислений и программирования в ограничениях. Для предложенных полимодельных комплексов разработаны методы для оценки их качества [13]. Создание таких моделей обеспечило возможность перехода от традиционного мониторинга к интеллектуальному.

Однако, в настоящее время, при необходимости наблюдения за большим числом разнородных взаимодействующих между собой объектов, в условиях, когда информация об объектах ограничена, возникают трудности оперативного построения, перестройки и использования имеющихся моделей, что сказывается на эффективности решения прикладных задач.

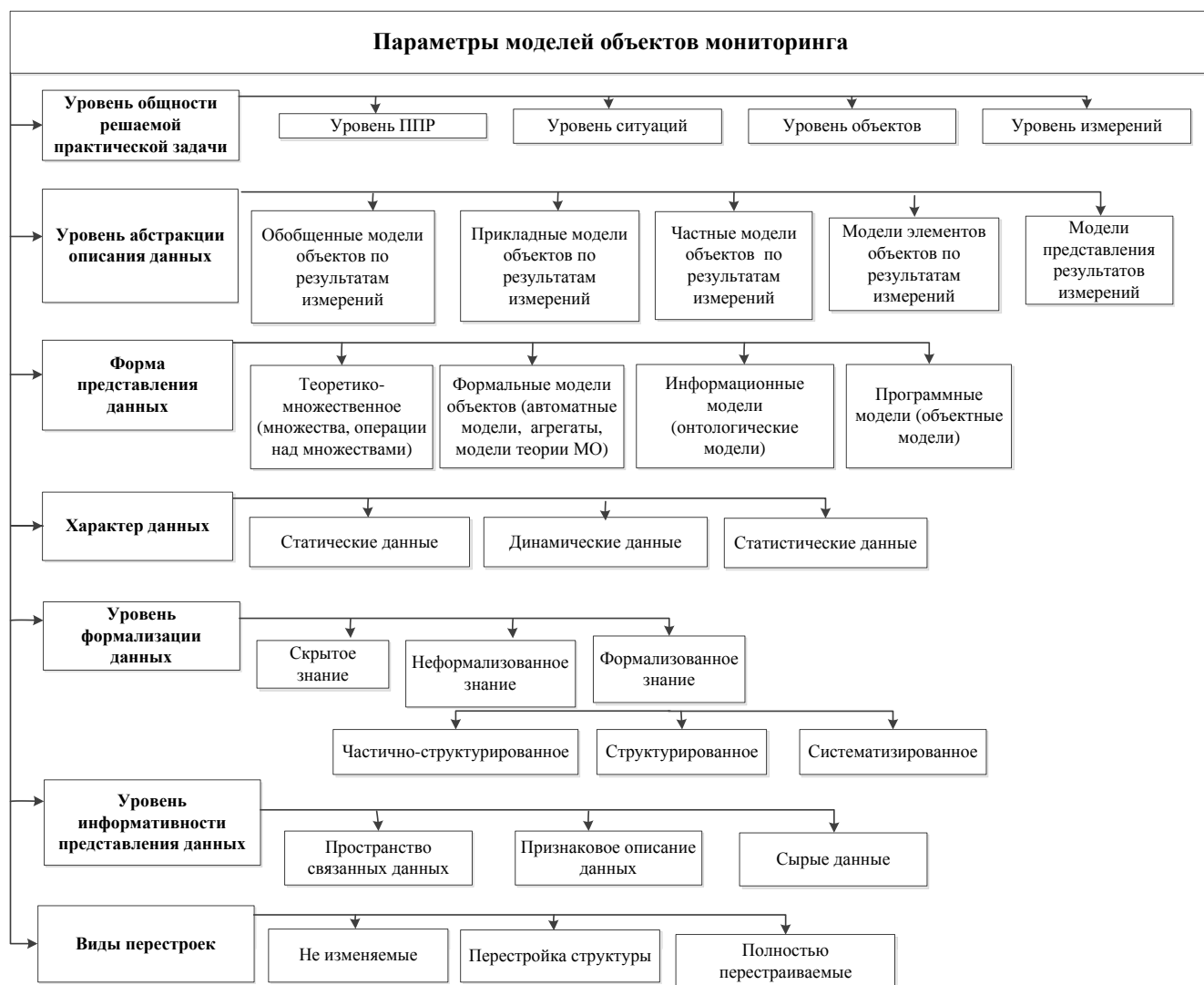


Рисунок 1.2 – Параметры моделей объектов мониторинга

### 1.3.2. Анализ методов построения моделей объектов мониторинга

Для построения моделей объектов мониторинга необходимы модели и методы, способные частично имитировать деятельность человека. Такие модели и методы разрабатываются для когнитивных систем, которые развиваются в трех направлениях: эмерджентного, символического и гибридного [14]. Для первого типа характерно использование продукционных и графовых моделей знаний. Второй тип предусматривает применение ассоциативных нейросетевых методов, а также других конкурентных моделей и методов. Гибридные архитектуры строятся на совместном использовании моделей знаний и нейросетевых методов. Они ожидаемо обладают наиболее широкими возможностями.

*Эмерджентные модели и методы.* Эмерджентные модели и методы разрабатываются на принципах биоподобия. Наиболее часто используются нейросетевые модели с ассоциативной обработкой сигналов. Нейросетевые модели и методы хорошо себя зарекомендовали, прежде всего, в распознавании образов в высокоразмерных данных, в обучении с подкреплением и в реализации ассоциативной памяти. Примерами известных эмерджентных когнитивных

нейросетевых решений являются: иерархическая темпоральная память (Hierarchical Temporal Memory (HTM)) [15]; система DesTIN [16], содержащая иерархическую сеть восприятия, подобную HTM, но более функциональную, и согласованную с ней иерархическую сеть, отвечающую за действия и их подкрепление; интегрированная биологически основанная архитектура (Integrated Biologically based Cognitive Architecture (IBCA)) [17], представляющая собой крупномасштабную эмерджентную систему, которая направлена на моделирование распределенной обработки информации в мозге. Она используется для симуляции различного психофизиологического и психолингвистического поведения человека; мобильное адаптивное устройство с нейронной организацией (Neurally Organized mobile Adaptive Device (NOMAD)) [18], основанное на нейронном дарвинизме Д. Эдельмана; системы, создаваемые на нейроморфных процессорах TrueNorth [19] и другие [20]. Однако пока не показано, как достичь в нейросетевых машинах функций высокого уровня, таких как абстрактное мышление, сложная осознанная обработка информации и другие, используя чисто нейросетевой подход, хотя потенциальные возможности у этого подхода чрезвычайно велики. Наиболее подходящими для решения задач построения моделей объектов мониторинга в рамках эмерджентного (нейросетевого) подхода являются рекуррентные нейросетевые структуры с ассоциативной обработкой информации. Необходимо, чтобы они позволяли не только оперативно, но и глубоко обрабатывать воспринимаемые сигналы и вырабатывать решения по их обработке. Однако из-за несовершенства моделей рекуррентных нейронных сетей совместить в них выполнение требований по оперативности и глубине ассоциативной обработке информации до последних лет не удавалось. Все известные до этого времени ассоциативные нейросетевые структуры [14]–[26] слабо приемлемы для глубокой и оперативной обработки непрерывных потоков разнородных сигналов. В частности, рекуррентные нейронные сети Хопфилда, Коско позволяют глубоко, но не оперативно обрабатывать информацию. Известные рекуррентные нейронные сети реального времени (RMLP, RTRN, сеть Эльмана) [23] реализуют быструю, но не глубокую ассоциативную обработку. Все эти нейронные сети в той или иной мере позволяют связывать обрабатываемые сигналы [21]–[23], [27], но не способны реализовать полноценное ассоциативное мышление. Некоторые новые решения, полученные в последние годы [28]–[33], позволяют надеяться на прорыв в этой области. Их дальнейшее развитие предусматривает поиск новых методов многоуровневого нейросетевого кодирования, принципов, методов и технологий связывания обрабатываемых сигналов в рекуррентных нейронных сетях, а также поиск принципиально новых нейросетевых структур. При этом требуются исследования вопросы управления ассоциативными многоуровневыми взаимодействиями разнородных обрабатываемых сигналов, а также обучения ассоциативных машин. Отдельно стоят вопросы исследования их устойчивого функционирования и технологий их программной и аппаратной

реализации. Решение перечисленных задач может потребовать длительного времени и усилий многих исследовательских коллективов.

*Символические модели и методы.* Наибольшее внимание длительное время уделялось символическим искусственным системам. Удалось создать ряд практически действующих, но слишком узко специализированных когнитивных символических машин. Обзор их можно найти в [14]. К общим недостаткам систем этой группы относятся низкий уровень и сложность их обучения, слабая креативность, отсутствие ассоциативного запоминания информации и другие [34], [35]. Имеются отдельные решения, ориентированные на применение парадигмы слияния данных [36], [37]. Разработанные системы применяются для решения четко обозначенных частных творческих задач. Ни одна из известных систем этого типа не является когнитивной в широком смысле.

Проведенный анализ показал, что существующие ИС, использующие когнитивные архитектуры, могут частично имитировать интеллектуальную деятельность человека за счет воспроизведения отдельных эвристических алгоритмов. Для построения моделей объектов мониторинга их способности оказываются недостаточны. Кроме того, когнитивные системы, по большей части, продолжают оставаться только экспериментальными системами. Несмотря на это, символические модели и методы могут иметь существенное развитие с переходом на многоуровневую обработку информации.

Для построения моделей объектов предлагается развитие символического подхода, в частности, систем, ориентированных на многоуровневое слияние данных, информации и знаний. Новые возможности по созданию систем могут быть обеспечены за счет применения аппарата синтеза при построении моделей объектов.

#### **1.4. Анализ моделей, методов и средств предварительной обработки данных мониторинга**

Основной интерес при построении моделей объектов мониторинга представляют модели и методы обработки данных, позволяющие описывать объекты на основе исходных потоков данных, поступающих от наблюдаемых объектов. Модели данных включают множество информационных элементов, между которыми установлены связи. Их построение обеспечивается за счет многократных переходов между различными представлениями данных. Каждое представление данных соотносится с некоторым уровнем модели.

Переходы между представлениями сопровождаются операциями трансформации данных. Понятие трансформации является заимствованным для области обработки данных. Существенное развитие теория трансформации получила в работе Дж. Лэнда [38]. Под трансформацией данных обычно понимается преобразование одного элемента в другой, выполняемое по некоторому правилу. В результате применения трансформаций могут формироваться новые информационные элементы или изменяться существующие.

Для определения трансформаций данных при построении моделей объектов необходимы формальные модели представления данных, а также методы и средства для их построения.

#### 1.4.1. Модели представления данных

Построение моделей представления данных предусматривает определение признакового пространства и описание в нем данных. Данные могут представляться в исходных и различных перестроенных признаковых пространствах. Перестройка признаковых пространств выполняется в нескольких случаях. Перестройка требуется при переходе между временным, частотным и частотно-временным пространствами. Также перестройкой признакового пространства сопровождаются переходы от исходных данных к их более емким представлениям. В ряде случаев, возникает необходимость в переходах между представлениями на одном уровне. Переходы выполняются за счет преобразования признакового пространства. Преобразование может быть направлено как на расширение, так и на сужение признакового пространства.

В исходном пространстве данные описываются значениями измеряемых параметров, в перестроенных - векторами характеристик.

Обычно выделяются три типа представлений [39]: не адаптивное, адаптивное, основанное на моделях. Для построения неадаптивных представлений данных применяются методы спектрального анализа, дискретного преобразования Фурье, дискретного вейвлет преобразования [40]. Достаточно широкое распространение получило представление, основанное на кусочно агрегативной аппроксимации, предложенной Е. Кеогхом (Piecewise Aggregate Approximation, PAA) [41]. Такая аппроксимация позволяет перейти к символьному представлению временных рядов (Symbolic Aggregate Approximation, SAX) [42].

Имеются многие специализированные преобразования, результаты которых могут применяться при построении векторов характеристик. К ним относятся сингулярное разложение (Singular value decomposition, SVD), разложение на эмпирические моды (Empirical mode decomposition, EMD) и другие.

Для адаптивного представления данных строятся аппроксимирующие функции. При аппроксимации могут рассматриваться полиномы различных степеней. Как правило, при аппроксимации данные представляются в виде множества фрагментов, для каждого из которых строится своя модель. В зависимости от выбранной степени полинома может строиться кусочно постоянная (Piecewise Constant Approximation, PCA), кусочно линейная аппроксимация (Piecewise Linear Approximation, PLA). При таком подходе свойства модели в значительной степени зависят от вектора переходных точек модели (Change point). Переходными являются точки, в которых происходит смена модели [43], [44]. К адаптивным моделям также относятся модели, формируемые с использованием шейплетов (Shapelets) [45], [46].

Построение представлений на основе моделей предполагает восстановление моделей по данным. В [47] рассматривается модель данных, построенная на основе вычисления статистических характеристик. В [48] в качестве моделей предложено использовать скрытые модели Маркова (Hidden Markov Model, НММ). Значительное распространение получили регрессионные модели как линейные, так и не линейные [49].

Для большинства моделей разработаны критерии для их оценки и методы, позволяющие подбирать параметры моделей. Для оценки линейных регрессионных моделей имеются информационные критерии Акаике (Akaike Information Criterion, AIC), Шварца (Schwarz Criterion), Ханнана-Куинна (Hannan-Quinn Criterion) и другие. Для моделей разрабатываются методы, позволяющие подбирать их параметры. Для подбора параметров регрессионных моделей применяются методы наименьших углов (Least angle regressions, LARS), метод Лассо (Least Absolute Shrinkage Selection Operator, LASSO), метод эластичных сетей (Elastic net estimator, NET) и т.д.

Для представлений данных разработаны меры, позволяющие их сравнивать. Сравниваться могут вектора характеристик и модели данных [50]. Для преобразования представлений данных в признаковых пространствах разработаны методы среди которых методы многомерного шкалирования, главных компонент [51].

#### **1.4.2. Методы связывания данных**

В [5] под связыванием понимается выявление, совместное использование, установление связей между элементами данных, информации, знаний. Связывание данных предполагает создание связей между элементами из разных источников на основе общих свойств, присущих этим элементам. Такие связи могут устанавливаться как на уровне крупных структур, так и на микроуровне. Связи могут быть вертикальными и горизонтальными. К горизонтальным относятся связи, устанавливаемые между элементами одного уровня, к вертикальным - между различными уровнями. Для установления горизонтальных связей применяются методы выявления зависимостей в данных, для вертикального связывания применяются модели и методы слияния данных. В результате связывания происходит обогащение данных, обеспечивается возможность их использования машинами.

*Модели и методы выявления зависимостей в данных.* Классические модели и методы выявления зависимостей между элементами данных лежат в области статистического анализа. Так, повсеместно используются регрессионные модели и методы их восстановления. Разработано значительное число модификаций регрессионных моделей как для одномерного (AR, MA, ARMA, Autoregressive Integrated Moving Average (ARIMA), Autoregressive Conditional Heteroskedasticity (ARCH), Season Autoregressive Integrated Moving Average (SARIMA), Autoregressive Fractional Integrated Moving Average (ARFIMA), ARIMAX), так и для



многомерного (Vector Autoregressive Models (VAM), Vector Moving Average Models (VMAM), Vector Autoregressive--Moving Average Models (VAMAM), VARMA, VARIMA, VARIMAX) случаев [52].

Для выявления более сложных видов зависимостей, в том числе, неочевидных косвенных зависимостей используются методы и средства, относящиеся к области ИИ [53], [54]. Эти модели, методы и средства формируют основу аппарата интеллектуального анализа данных, в частности Data Mining (DM), а также аппарата машинного обучения (Machine Learning, ML) [55], [56]. Можно выделить следующие основные группы методов: предварительная обработка данных (Preprocessing), кластеризация (Clustering: Representative-based Clustering, K-means, Expectation-Maximization Clustering, Hierarchical Clustering, Density-based Clustering, Spectral and Graph Clustering), классификация и прогнозирование (Classification and Prediction: Probabilistic Classification, Decision Trees, Linear Discriminant Analysis, Support Vector Machines), поиск шаблонов (Pattern Mining: Itemset Mining, Sequence Mining, Graph Pattern Mining, Pattern and Rule Assessment), поиск ассоциативных правил (Association Mining) [53]–[60]. Последние две группы методов ориентированы на выявление логических зависимостей в данных. Анализ этих групп методов приведен в [61], [62].

Для связывания данных могут использоваться другие подходы, в частности, аппарат анализа формальных понятий [63], [64], а также нейронные сети [23].

Среди достижений отечественных ученых в области связывания данных значительный интерес представляют классические направления ИИ [54], а также новые направления, в том числе, получившие развитие в научной школе Санкт-Петербургского института информатики и автоматизации РАН. Направления представлены в работах Александрова В.В., Городецкого В.И., Смирнова А.В., Тулупьева А.Л., Хованова В.В и других. В области классификации, распознавания образов и прогнозирования важные результаты для построения моделей объектов получены в Институте математики СО РАН под руководством Загоруйко Н.Г. [65]–[68].

В последние годы зарубежные исследователи в области обработки и анализа данных существенное внимание уделяли вопросам анализа временных рядов. К одной из первых значимых работ в области интеллектуального анализа временных рядов можно отнести работу Повинелли [69]. За ней последовал целый ряд исследований, благодаря которым интеллектуальный анализ временных рядов выделился в отдельное направление. В рамках этого направления решаются задачи сегментации и классификации временных рядов, поиска временных шаблонов и другие. Многие интересные результаты в области сегментации временных рядов получены специалистами NASA, в частности Jeffrey D. Scargle [70], [71].

При классификации временных рядов в [72] предлагается исходить из оценок схожести описывающих их кривых, а также схожести наблюдаемых во временных рядах изменений. При классификации временных рядов во времени хорошо себя зарекомендовали такие классификаторы, как 1-NN классификатор [73]. Имеются классификаторы, основанные на вычислении характеристик временных рядов инвариантных ко времени. Сравнение временных рядов на основе оценки их изменений требует построения моделей временных рядов (НММ, ARMA и др.) [74].

Среди методов классификации временных рядов также хорошо себя зарекомендовали методы, разработанные Е. Кеогхом [46]. Эти методы основаны на построении символьного представления временных рядов. Имеются примеры их успешного применения при решении прикладных задач [50].

Для выявления шаблонов в поведении временных рядов используются методы, основанные на применении темпоральной логики [75], [76]. Помимо этих методов, имеются методы, ориентированные на поиск мотивов во временных последовательностях [77].

Результаты, полученные за последние годы, в основном, развивают ранее предложенные модели и методы.

В целом, анализ существующих решений по связыванию данных показал, что имеется значительное число моделей и методов, позволяющих выполнять обработку данных, необходимую для построения моделей объектов по результатам наблюдений. Однако, имеется ряд факторов, затрудняющих их использование. Одним из значимых факторов является ограниченная область применения большинства методов. Каждый из методов может оказаться востребованным в некоторых условиях. Нарушение условий применения методов приводит к получению недостоверных результатов. Другим фактором является сложность применения методов. Необходима настройка многих параметров для получения требуемых результатов.

*Модели и методы слияния данных.* В соответствии с классическим определением слияние данных – это процесс объединения, сравнения, поиска взаимосвязей в данных и информации, поступающих от одного или нескольких источников, имеющий конечной целью определение ситуации, ее оценку с точки зрения наличия угроз. Это непрерывный процесс, который отслеживает развитие ситуации в динамике [78]. Существует более короткое определение термина, в соответствии с которым слияние данных – это процесс, отвечающий за организацию, комбинирование и интерпретацию данных, поступающих от одного или нескольких источников [79]. Анализ используемой в настоящее время терминологии различными авторами показал, что она достаточно противоречива [80]–[82]. Так, например, часто конфликты возникают между понятиями поглощение, слияние, соединение, комбинация, сочетание. Часто плохо разделимыми оказываются понятия слияния и интеграции. Термин

слияние может использоваться не только применительно к данным. Он также часто встречается при описании процессов преобразования информации, знаний. При этом он несколько меняет свое значение. Под слиянием информации понимается процесс объединения (комбинации) информации в новый набор информации, которая отличается от исходной меньшей неопределенностью. Под слиянием знаний понимается процесс порождения качественно новых знаний на основе имеющейся совокупности информации и знаний с применением широкого набора методов и средств искусственного интеллекта. Другие варианты определения этих терминов, а также связанных с ними терминов, предложены в [79]–[84].

В рамках общей проблемы слияния данных значительное внимание уделяется слиянию данных от датчиков.

Алгоритмы слияния данных от датчиков разделяются на 3 категории:

- алгоритмы, ориентированные на позиционное слияние;
- алгоритмы, ориентированные на установление идентичности объектов;
- вспомогательные алгоритмы, ориентированные на обработку результатов измерений.

Выделяют несколько типов слияний данных от датчиков. К ним относятся слияние атрибутов (Fusion of Attributes), слияние результатов анализа (Fusion of Analysis), слияние представлений (Fusion of Representation) и др. Слияние данных, как правило, предусматривает построение моделей, описывающих поведение данных. Если модели поведения неизвестны или ошибочны, то процесс слияния данных может не давать требуемых результатов.

Для решения задач слияния разработано несколько подходов: прямое слияние, последовательное слияние, смешанное (векторное) слияние. При слиянии используются различные модели слияния данных. Модели слияния данных классифицируются по трем признакам: тип модели, область применения модели, количество уровней модели. По типу различают функциональные, информационные, процессные, ролевые и смешанные модели. В зависимости от области применения выделяются модели с широкой областью применения и модели, ориентированные на решение отдельных задач. Модели с широкой областью применения могут использоваться в нескольких предметных областях. В зависимости от числа уровней выделяются одноуровневые модели, модели, имеющие несколько уровней, и модели, имеющие все уровни.

Общая характеристика моделей приведена в [81]. В основном, модели представлены на абстрактном уровне, реже доведены до конкретных узкоспециализированных решений (модели OODA [79], водопада [85], Омнибус [86], Рассмунсена [87], Дезарати [88], [89]). Это существенно осложняет их использование при обработке данных.

Некоторые модели, методы и алгоритмы слияния данных приведены в [90]–[92].

Результаты проведенного анализа моделей слияния данных показывают, что наиболее приемлемой для обработки данных наблюдений является JDL модель [93]–[95]. Она напрямую не связана ни с одной из предметных областей. JDL модель представлена на рисунке 1.3. Модель включает 6 уровней. JDL модель – это функциональная модель. Фактически, эта модель определяет то, “что делается”, а не то “как делается”. Прослеживается явная тенденция к наполнению каждого из уровней конкретной прикладной функциональностью. JDL модель не рассматривается как процессная модель или как модель технической архитектуры. Реальные модели и процессы слияния данных (Data Fusion Systems), построенные на ее основе, являются достаточно сложными.

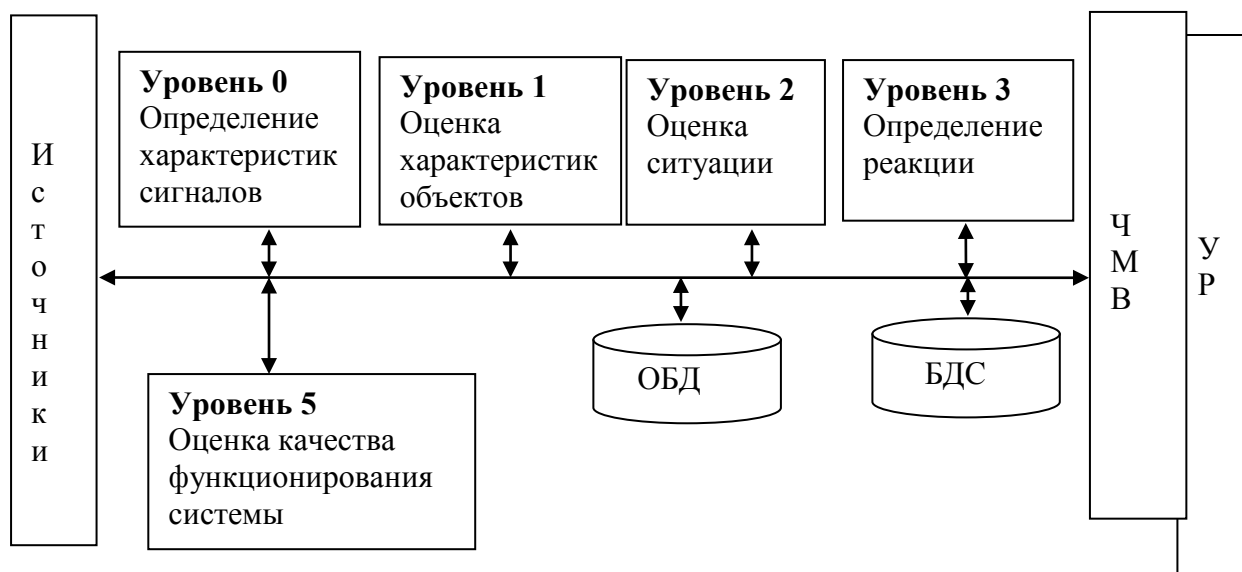


Рисунок 1.3 – JDL модель:

ОДБ – оперативная база данных, БДС – база данных системы, ЧМВ – человеко-машинное взаимодействие, УР– управляющая реакция

Предполагается, что в рамках JDL модели реализуется двунаправленный иерархический процесс. Переходы по уровням модели осуществляются снизу вверх и сверху вниз – от нулевого уровня к третьему уровню и в обратном направлении. Движение снизу вверх предполагает обработку данных, затем выделение из них знаний и принятие решений.

В дополнение к этой модели применяют визуальную модель слияния данных [96], позволяющую отражать информацию и результаты ее многоуровневой обработки.

Имеются отдельные работы, в которых рассматриваются технологии реализации многоуровневых моделей обработки данных [97]–[99]. Однако, большинство из них являются специализированными [100], [101].

В целом, результаты анализа показали, что на сегодняшний день отсутствуют модели и методы, позволяющие решать задачи слияния данных при обработке данных мониторинга.

### 1.4.3. Средства обработки данных мониторинга

При построении моделей объектов мониторинга необходимы модели, содержащие данные о моделях и методах обработки, а также о критериях оценки формируемых результатов. Такие модели относятся к моделям области обработки и анализа данных. Они должны позволять осуществлять обоснованный выбор методов обработки при известных исходных данных и требованиях к результирующим представлениям.

На текущем этапе развития моделей обработки данных они представлены, в основном, в виде таксономий алгоритмов машинного обучения [102]–[104].

Таксономии проработаны для значительной части основных групп методов статистической обработки данных и машинного обучения. Разработаны таксономии методов предварительной обработки данных [105]. Таксономия методов кластерного анализа предложена в [106], методов классификации - в [107]–[109]. В [110] предложена таксономия методов классификации временных рядов, проведено сравнение производительности методов на основе результатов большого числа экспериментов. Таксономия включает наиболее известные методы классификации. Однако, в данной таксономии не рассмотрены методы классификации на основе моделей. Методы классификации на основе моделей, а также методы, основу которых составляют модели и методы глубокого обучения, представлены в [111]. Сравнительные оценки методов из [110], [111], полученные на стандартных тестовых наборах (архив UCR/UEA [112], [113]), приведены в [114]. В [110] также приведены типовые модели представления временных рядов. В [115] представлено значительное число мер схожести, позволяющих сравнивать временные ряды.

Имеется ряд отдельных работ, направленных на исследование критериев оценки результатов обработки [116]. Информационные критерии рассмотрены в [117].

Приведенные таксономии не ориентированы на решение задач выбора методов при обработке данных. Между элементами моделей устанавливаются отношения "часть-целое". Другие отношения практически не рассматриваются. В результате, модели не содержат достаточно информации для выбора методов при обработке данных.

Вопросы выбора методов частично рассмотрены в [118], [119]. Ряд решений предложен в [120]–[127]. Однако, эти решения имеют ограниченную область применения.

В отдельных работах рассматривается проблема построения процессов обработки данных. Для построения процессов предлагается создавать интеллектуальных ассистентов [121]. Также предложены решения, основанные на применении семантических моделей [128]–[131].

К последним работам в области построения моделей обработки данных относятся [132]–[136]. В них представлены семантические модели [137], позволяющие выполнять обработку

больших объемов данных с использованием методов машинного обучения. Например, разработаны модели и методы для анализа данных Твиттера [132], для построения моделей индивидуального обучения [134]. Состав разработанных новых моделей показывает, что предложенные решения продолжают оставаться узконаправленными, имеют ограниченные возможности по построению процессов обработки.

В целом, проведенный анализ показал, что для построения процессов обработки данных мониторинга имеется необходимость в развитии существующих решений. При построении моделей объектов формулируются различные задачи обработки, что требует моделей обработки данных с достаточно широкой областью применения. Кроме того, особенности данных мониторинга требуют уделять отдельное внимание вопросам их предобработки. Также требуется наличие механизмов, позволяющих выбирать методы обработки исходя из характеристик обрабатываемых данных. Для этого необходимо иметь более широкие возможности для оценки получаемых результатов.

### **1.5. Анализ методов синтеза моделей объектов, процессов и программ мониторинга**

К настоящему времени разработано значительное число методов и моделей синтеза. Их детальный анализ приведен в статье автора [138]. Классификация методов и моделей синтеза приведена на рисунке 1.4. Она построена исходя из решаемой задачи построения моделей объектов мониторинга. Эта классификация позволяет определить место разрабатываемых решений в рамках существующих методов и моделей.

При синтезе моделей объектов мониторинга необходимо синтезировать формальные структуры и соответствующие им исполняемые структуры.

*Модели и методы синтеза формальных структур.* Среди существующих моделей и методов синтеза формальных структур наибольший интерес представляют подходы к синтезу процессов и программ, рассмотренные в [139]–[150]. Синтез формальных процессов и программ предусматривает построение их вариантов, оценку эффективности, выбор целесообразных вариантов. При этом требуется обеспечить результативность синтезируемых программ. Результативными являются программы, которые завершают работу за конечное число шагов хотя бы на одной интерпретации значений исходных данных. На других интерпретациях работа программ должна завершаться за конечное число шагов с выдачей результата об успешном или безуспешном решении. Существование программ на заданном множестве условий, которые связывают исходные данные с планируемым результатом, доказываются с применением методов резолюций [139], [143]. Для извлечения программ из доказательств применяются методы обратного вывода [142].

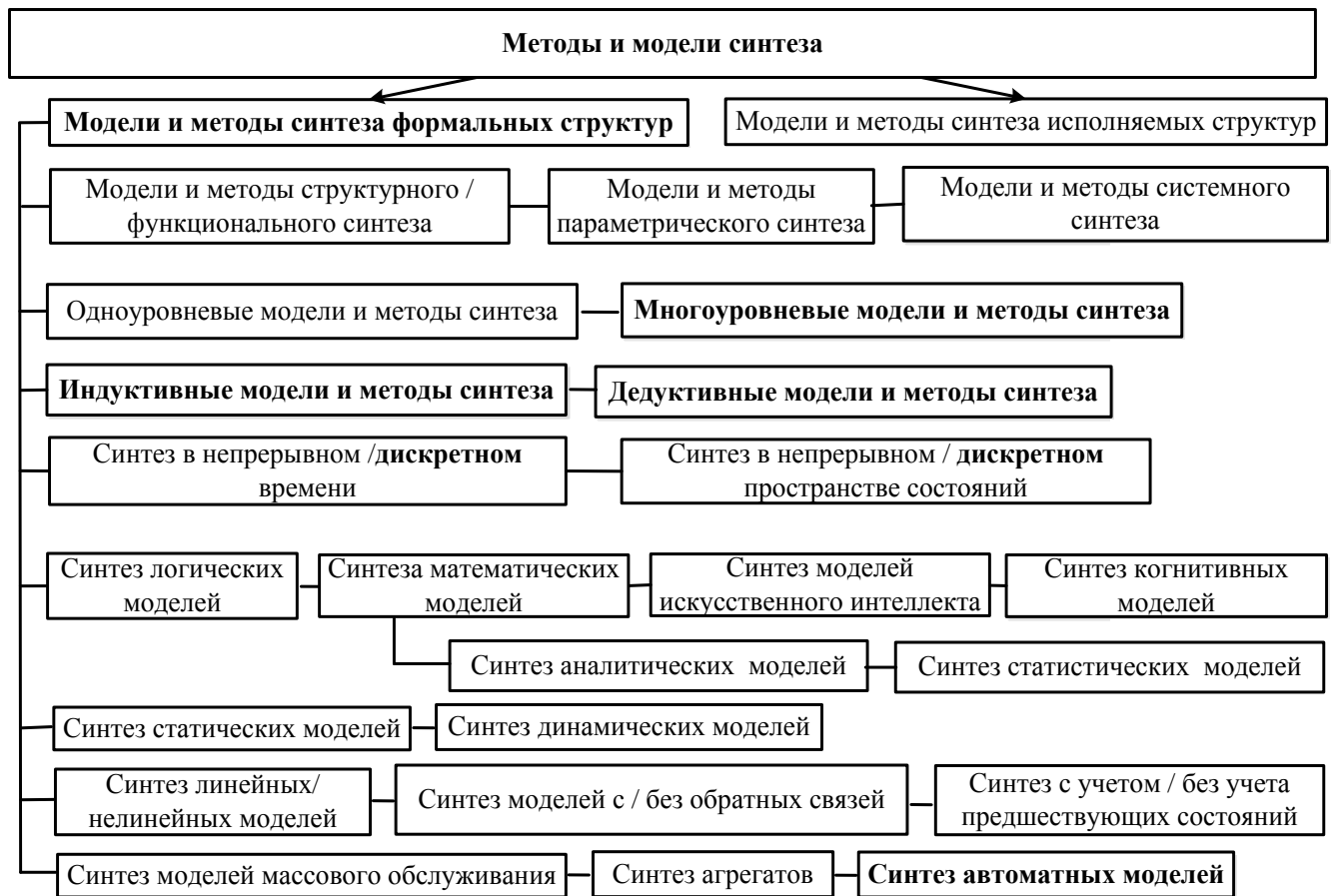


Рисунок 1.4 – Классификация методов и моделей синтеза

Для успешного решения задачи синтеза процессов и программ необходимо их формальное описание в виде набора условий (правил), связывающих различные данные. Формальное описание простых процессов и программ, как правило, не составляет проблемы. Однако, синтез реконфигурируемых процессов и программ в условиях изменяющегося контента и контекстов составляет более сложную задачу, поскольку возникает необходимость в использовании специальных правил, описывающих их возможные структуры.

Для уменьшения сложности синтеза обычно используют различные методы распараллеливания доказательства существования программ и извлечения их из выводов. Однако это не позволяет решить проблему сложности в полной мере. Необходимо выйти на существенно более низкий уровень сложности решения задач синтеза.

Один из возможных путей решения проблемы вычислительной сложности связан с использованием многоуровневого подхода, который был обозначен еще в 1983 году [151] и получил развитие в работах [152], [153]. Однако, до настоящего времени, многоуровневый подход к синтезу реконфигурируемых процессов и программ в должной мере не проработан. Для этого требуется разработать более эффективные модели и методы синтеза. При этом имеющиеся модели и методы синтеза одноуровневых моделей могут составить основу нового аппарата синтеза.

*Модели и методы синтеза исполняемых структур.* К настоящему времени широкое применение на практике получили модели и методы синтеза исполняемых программ [154], [155].

Когда говорят об автоматической генерации программ, то фактически речь идет о нескольких близких по постановке, но разных по способу решений и результатам задачам. К таким частным задачам относятся следующие задачи:

1. Построение исполняемой (интерпретируемой) программы по ее описанию на языке более высокого уровня:  $PRG_{i+1} \rightarrow PRG_i$  где  $PRG_i$  - описание исполняемой программы, а  $PRG_{i+1}$  - описание программы на более высоком уровне. Очевидно, что в данном случае речь идет о компиляторе или интерпретаторе. Вопросы построения компиляторов и интерпретаторов очень подробно исследованы в [156].

2. Построение исполняемой (интерпретируемой) программы по модели  $M_{i+1} \rightarrow M_i \rightarrow PRG$ , где  $M$  – модель системы, а  $PRG$  – требуемая программа. При этом могут быть использованы самые разные модели и способы их описания. Это могут быть, в частности, языки архитектурного описания либо доменно-ориентированные языки [157]. Такой подход положен в основу проектирования на моделях [139]. Он построен на механизме трансформации моделей. Трансформации моделей применяются при решении задач построения исполняемых структур по синтезированным формальным моделям.

На основе моделей могут генерироваться скрипты или исполняемые программы. Они могут отправляться для исполнения на устройства сбора данных и другие. Результатом генерации могут быть бизнес-процессы. Таким образом, в частности, работают семантические веб-сервисы [158]. Программы и процессы могут либо строиться полностью, либо могут модифицироваться их отдельные фрагменты. Для построения процессов и программ с изменяющимся поведением могут применяться механизмы бизнес-правил и бизнес-политик [159], [160]. Процесс генерации изменений может инициироваться оператором или некоторым внешним событием, связанным с изменениями в контенте или контексте поступающих информационных потоков.

В целом, анализ уровня проработанности проблемы синтеза показал, что имеется значительное число моделей и методов синтеза. Однако, возможности их применения для синтеза моделей объектов мониторинга достаточно ограничены. Основным препятствием является высокая вычислительная сложность одноуровневого синтеза формальных моделей.

Таким образом, требуется аппарат многоуровневого автоматического синтеза формальных моделей наблюдаемых объектов в виде относительно конечных операционных автоматов. Разрабатываемый аппарат должен обеспечивать возможность многоуровневого синтеза процессов и программ мониторинга.



### 1.6. Формулировка проблемы и логическая схема исследования

Проведенный анализ процессов построения моделей объектов мониторинга позволяет утверждать, что уровень развития теории синтеза во многом не удовлетворяет современным потребностям практики. На синтез этих моделей затрачивается много ресурсов, при этом получаемые модели объектов мониторинга не всегда отвечают предъявляемым им требованиям. Это может негативно сказываться на эффективности принимаемых управленческих решений и приводить к существенным материальным потерям.

Таким образом, остро назрела научная проблема разработки основ теории и методов многоуровневого автоматического синтеза автоматных моделей объектов мониторинга, отвечающих современным потребностям практики по снижению вычислительной сложности этого синтеза.

Цель решения этой проблемы - развитие теории многоуровневого автоматического синтеза моделей объектов мониторинга, кратно снижающей вычислительную сложность этого синтеза.

Логическая схема решения этой проблемы в диссертации отражена на рисунке 1.5.

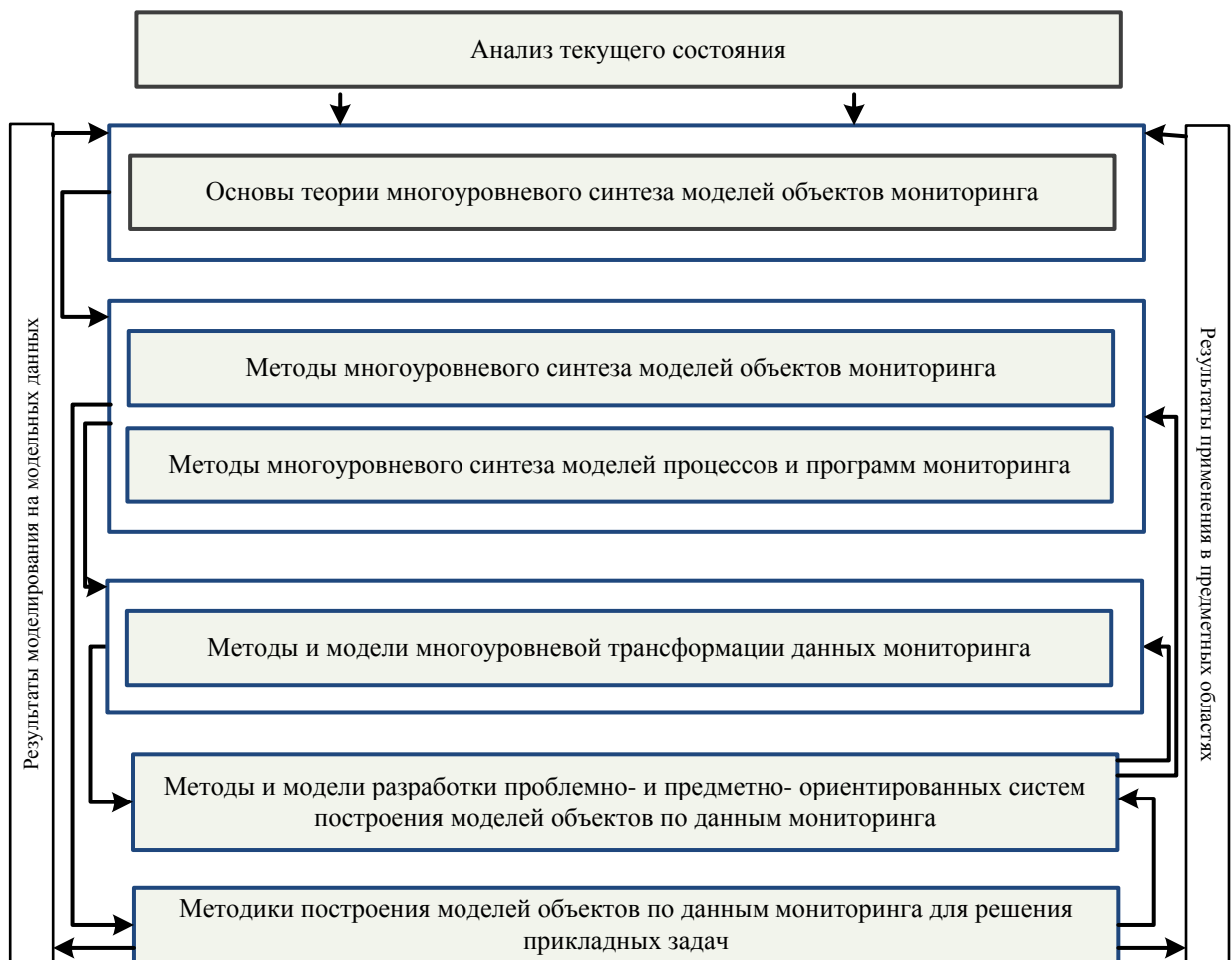


Рисунок 1.5 – Логическая схема исследования

Логическая схема предусматривала анализ текущего состояния исследуемой области, обоснование потребности в многоуровневом синтезе моделей объектов мониторинга в предметных областях.

Далее логическая схема предусматривала разработку основ теории многоуровневого синтеза моделей объектов, процессов и программ мониторинга, а также научно методологического аппарата, позволяющего синтезировать формальные модели. Для синтеза конкретных моделей предусматривалась разработка моделей и методов трансформации данных мониторинга. Разработанный аппарат синтеза положен в основу нового класса систем построения моделей объектов по данным мониторинга.

Для применения предлагаемого аппарата синтеза на практике предусматривалась разработка общей методики построения моделей объектов по данным мониторинга, а также частных методик для конкретных предметных областей. Результаты эксплуатации разработанных и внедренных систем обрабатываются и определяют направления дальнейшего развития предлагаемого многоуровневого синтеза.

### **1.7. Выводы**

1. Проведен анализ современных потребностей прикладных предметных областей в построении моделей объектов мониторинга при решении практических задач, включающих задачи прогнозирования и управления. Результаты анализа позволили выявить несоответствие имеющихся методов и средств предъявляемым к ним требованиям из-за высокой вычислительной сложности.

2. Определены основные особенности наблюдаемых объектов. Объекты имеют многоуровневую структуру, изменяющуюся во времени, реализуют многие целевые процессы. Для них характерно наличие сложных внутренних взаимосвязей. От объектов поступают большие объемы данных. Модели объектов описываются в дискретном времени и дискретном пространстве состояний. Определены основные классы задач синтеза моделей объектов мониторинга.

3. Определены условия построения моделей объектов мониторинга. Основным источником данных об объектах являются поступающие от них информационные потоки. Установлено, что объемы исходных данных непрерывно увеличиваются, однако доля используемых данных продолжает оставаться низкой. Выявлены основные проблемы, возникающие при обработке данных, определены свойства контента информационных потоков, рассмотрены возможные контексты обработки контента.

4. Проведен анализ известных моделей объектов, процессов и программ мониторинга и методов их построения. Установлено, что модели и методы разрабатываются в рамках эмерджентного, символического и гибридного подходов. Показано, что возможностей

существующих ИС оказывается недостаточно для эффективного построения моделей объектов мониторинга. Предложено развитие символического подхода, в частности, решений, основанных на многоуровневом слиянии данных. Новые возможности предложено обеспечить за счет развития аппарата синтеза.

5. Выполнен анализ моделей, методов и средств предварительной обработки данных мониторинга, направленной на подготовку данных для синтеза моделей объектов мониторинга. Рассмотрены модели представления исходных данных, а также методы и модели связывания данных, необходимые для их построения, дана оценка возможностям существующих средств построения процессов обработки данных. Результаты анализа показали, что существующие модели и методы могут позволить проводить предварительную обработку данных для синтеза моделей объектов, однако известные решения по построению процессов обработки данных требуют дальнейшего развития.

6. Выполнен анализ методов синтеза моделей объектов, процессов и программ мониторинга. Показано, что для синтеза моделей объектов мониторинга интерес представляют методы синтеза реконфигурируемых процессов и программ. Для описания программ с реконфигурируемой структурой используются относительно конечные операционные автоматы. Установлено, что основным ограничением для применения этих методов при синтезе моделей объектов является их высокая вычислительная сложность. Снижение вычислительной сложности возможно за счет использования многоуровневого подхода при синтезе. Однако, до настоящего времени, многоуровневый подход к автоматическому синтезу реконфигурируемых процессов и программ в должной степени не проработан.

7. Сформулирована актуальная научная проблема разработки основ теории и методов многоуровневого автоматического синтеза автоматных моделей объектов мониторинга, отвечающих современным потребностям практики. Разработана логическая схема для ее решения.

## **Глава 2. Основы теории многоуровневого синтеза автоматных моделей объектов мониторинга**

### **2.1. Концепция многоуровневого синтеза автоматных моделей объектов мониторинга**

В рамках концепции формулируются основные положения многоуровневого синтеза моделей объектов мониторинга и предлагается концептуальная модель. Концептуальная модель определяет модель общего процесса многоуровневого синтеза моделей объектов мониторинга. Реализация многоуровневого синтеза обеспечивается средствами абстрактного вычислителя.

В основе многоуровневого синтеза моделей объектов мониторинга лежат следующие концептуальные положения:

1. Синтез моделей объектов должен выполняться на многих уровнях. Это должно обеспечить снижение вычислительной сложности синтеза.

2. Синтезируемые модели должны являться формальными полностью перестраиваемыми моделями, представляемыми в виде автоматных моделей. Это должно обеспечить широкие возможности по автоматической перестройке полученных моделей.

3. Синтез моделей объектов мониторинга должен строиться на выявлении и многоуровневом связывании элементов данных между собой. Это должно обеспечить построение моделей, отражающих структурные, логические и иные взаимосвязи между элементами наблюдаемых объектов.

4. Синтез моделей объектов мониторинга должен осуществляться по результатам обработки потоков данных, поступающих от объектов и из иных источников. Это должно обеспечить построение моделей объектов, соответствующих текущему состоянию наблюдаемого объекта и условиям мониторинга.

5. Синтез моделей объектов мониторинга должен обеспечиваться гибкими программными системами. Гибкие системы построения моделей объектов должны описываться на модельном уровне, их реализация осуществляется за счет интеграции готовых компонентов. Это должно обеспечить надежность систем и их невысокую сложность разработки.

В развернутом виде положения сформулированы в [138], [161].

#### **2.1.1. Концептуальная модель многоуровневого синтеза автоматных моделей объектов мониторинга**

Структурная схема концептуальной модели многоуровневого синтеза автоматных моделей объектов мониторинга представлена на рисунке 2.1. Модель рассмотрена в статье автора [162].

Предлагаемая модель предусматривает, что имеются объекты реального мира, за которыми ведется наблюдение. Необходимо по результатам наблюдений построить модели

объектов, отвечающие требованиям конечных пользователей. Требования представляются в виде целевых моделей объектов.

Для наблюдения применяются существующие средства мониторинга. Кроме данных, относящихся непосредственно к наблюдаемому объекту, осуществляется сбор данных о состоянии внешней среды. Наблюдение за объектами ведется целенаправленно исходя из имеющихся о наблюдаемых объектах данных и целевых моделей.

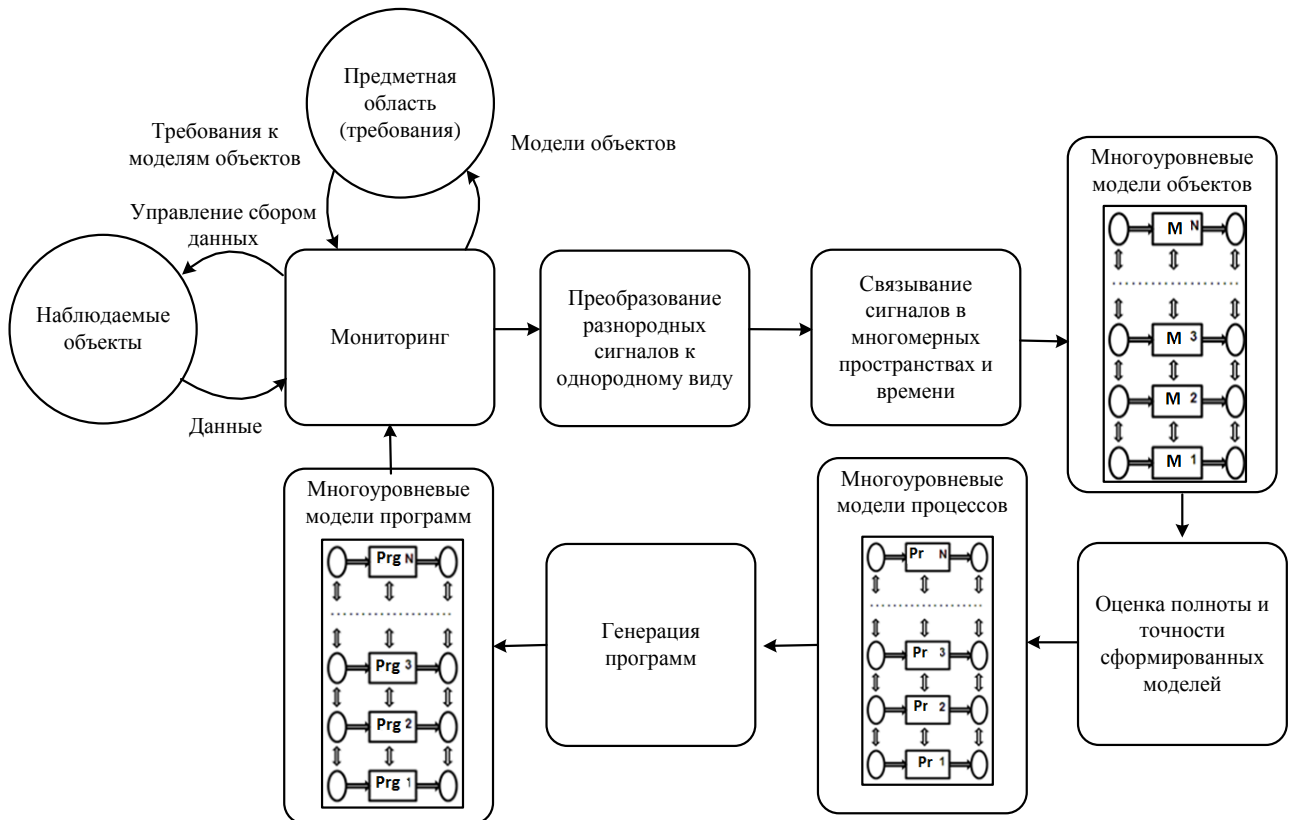


Рисунок 2.1 – Структурная схема концептуальной модели многоуровневого синтеза автоматических моделей объектов мониторинга

Основными понятиями, используемыми при многоуровневом синтезе моделей объектов мониторинга, являются: модели объектов мониторинга, модели процессов мониторинга, модели программ мониторинга.

Модели объектов мониторинга – это перестраиваемые многоуровневые изменяющиеся в пространстве и во времени структуры, представляющие объекты в виде множества взаимосвязанных информационных элементов.

Модели процессов мониторинга - это модели, определяющие перестраиваемые многоуровневые процессы сбора и обработки данных о наблюдаемых объектах.

Модели программ мониторинга - это программно-реализуемые структуры, транслируемые в программный код.

Процесс синтеза моделей наблюдаемых объектов разделяется на три этапа.

Этап 1. На первом этапе выполняется синтез моделей объектов по имеющимся данным за счет преобразования этих данных и их связывания. На этом этапе значительную роль играют модели и методы обработки данных. Синтезированные модели оцениваются. Оценка предполагает определение параметров моделей и сравнение моделей с целевыми моделями. В случае, если модели соответствуют предъявляемым требованиям, поставленная задача синтеза является решенной. Однако, обычно синтезированные модели оказываются не полными и не отвечают предъявляемым к ним требованиям. В таких ситуациях формулируются и решаются задачи сбора недостающих данных. Для этого синтезируются новые процессы мониторинга или перестраиваются существующие. Синтез процессов мониторинга осуществляется на втором этапе.

Этап 2. Процессы мониторинга синтезируются на основе отличий моделей, построенных по полученным данным, и целевых моделей. Определяются данные, которые должны быть получены для перестройки имеющихся моделей таким образом, чтобы они отвечали требованиям. Программы мониторинга строятся на третьем этапе.

Этап 3. Программы мониторинга синтезируются на основе построенных процессов мониторинга. Вначале строятся формальные программы, а затем генерируются исполняемые. При генерации исполняемых программ формальные программы транслируются на языки программирования. Программы исполняются на конечных устройствах. В результате осуществляется получение новых данных.

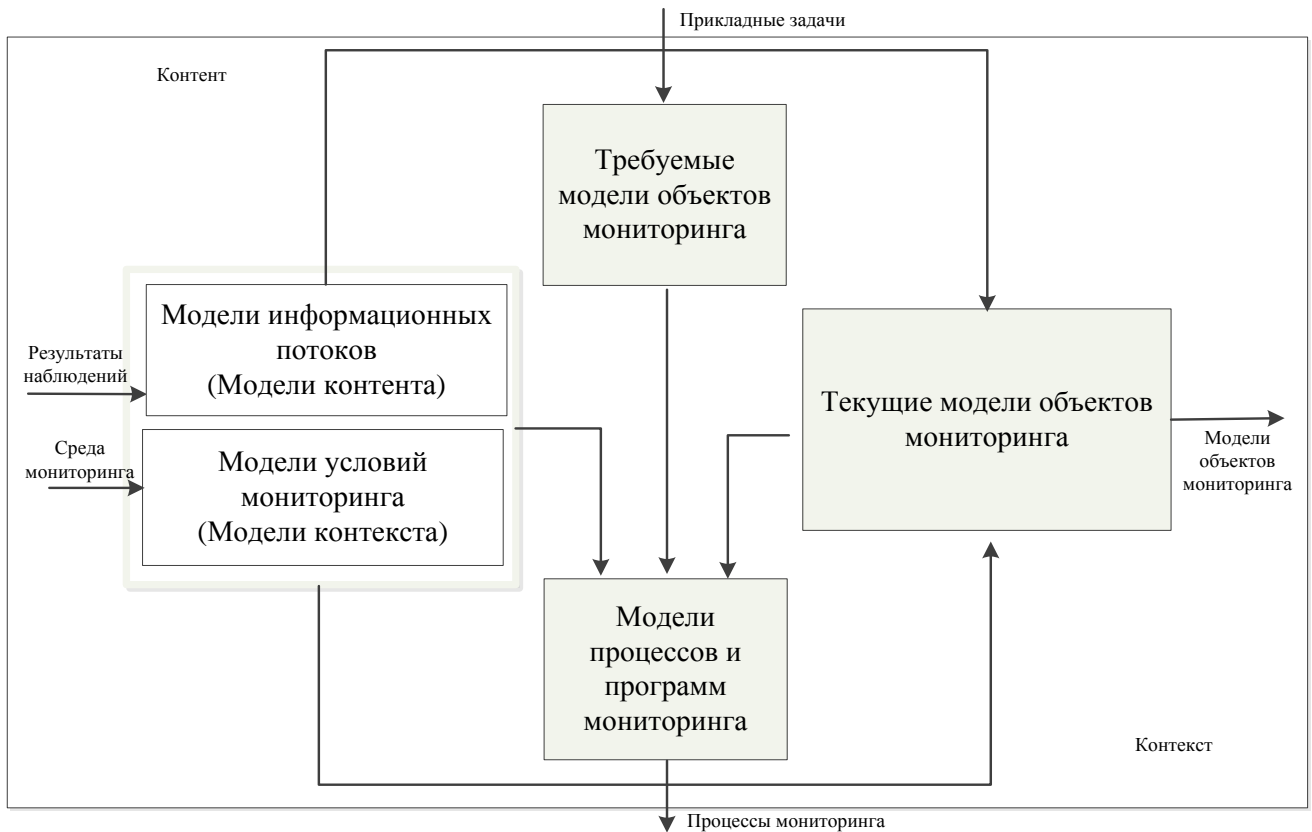
После получения новых данных цикл синтеза повторяется.

Таким образом, предлагаемая модель многоуровневого синтеза автоматных моделей объектов определяет полный цикл синтеза моделей наблюдаемых объектов, включающий синтез процессов и программ мониторинга для сбора необходимых об объектах данных.

### **2.1.2. Машина многоуровневого синтеза**

Многоуровневый синтез моделей объектов обеспечивается средствами машины многоуровневого синтеза (ММС). Машина рассмотрена в статьях автора [10], [11], [161]. Структура машины приведена на рисунке 2.2. Предлагаемая машина представляет собой абстрактный вычислитель, обеспечивающий решение задач синтеза моделей наблюдаемых объектов и необходимых для этого процессов и программ мониторинга. При изменении потоков данных или условий мониторинга модели в ММС перестраиваются.

На основе данных о наблюдаемых объектах в ММС формируются модели контента. Они представляют собой структуры, которые отражают изменения содержания потоков в пространстве и во времени. При их построении решаются задачи прямого и обратного анализа данных. Данные об условиях мониторинга представляются в виде моделей контекста.



Рисунке 2.2 – Структура машины многоуровневого синтеза

На основе моделей контента и контекста синтезируются модели объектов мониторинга. Они содержат информацию о состояниях этих объектов, связях между состояниями, условиях существования связей, а также другую информацию, характеризующую наблюдаемые объекты и их поведение. При синтезе моделей обеспечивается возможность синтеза их структур, а также наполнение этих структур. Модели синтезируются с применением индуктивного и дедуктивного подходов.

При построении моделей объектов синтезируются модели процессов мониторинга. Синтез моделей процессов предусматривает, что выполнено построение моделей по имеющимся данным и заданы требуемые модели. Синтез моделей процессов основан на методах дедуктивного синтеза. Доказывается возможность перехода от имеющихся моделей к требуемым. Наблюдение за объектами, которое производится в соответствии с синтезированными процессами, обеспечивает поступление новых данных на вход ММС.

Модели в ММС характеризуются рядом параметров, в частности, уровнем общности решаемых задач, уровнем абстракции описания контента, формой представления контента и другими.

Формально предлагаемую машину можно представить как

$$MAS = MAS(AVT_1, AVT_2, AVT_3, AVT_4); \quad (2.1)$$

где AVT- модели, формализующие отдельные блоки машины автоматического синтеза (Machine for Automatic Synthesis, MAS).

Рассматриваемая машина автоматического синтеза предполагает использование аппарата относительно конечных операционных автоматов. Это позволяет строить и перестраивать модели наблюдаемых объектов в пространстве их состояний и в функциональном пространстве. При построении моделей процессов определяются последовательности переходов, которые позволяют перевести модели объектов из исходных состояний  $\bar{d}_r$  в требуемые  $\bar{d}_s$ :  $F^b: \bar{d}_r \rightarrow \bar{d}_s$ , где  $F^b$  - множество функций переходов.

## 2.2. Обобщенная формулировка задачи многоуровневого синтеза моделей объектов мониторинга

В общем виде задача многоуровневого синтеза моделей объектов мониторинга формулируется следующим образом [10], [11], [138]. Требуется путем синтеза  $S$  найти модель  $B_0$  на заданный момент времени  $t$ , при которой достигается экстремум основного показателя эффективности многоуровневого синтеза  $W$  при ограничениях на вспомогательные показатели  $M$ :

$$B_0(t) \in \underset{v \in \Lambda}{\text{Arg extr}} W(S(B_v, t), M(B_v, t), t = \overline{1, T}) \quad (2.2)$$

$$M(B_v, t) = F(M_1(B_v, t), M_2(B_v, t), \dots, M_n(B_v, t)); \quad (2.3)$$

$$\Phi(M_1) \in E_1, \Phi(M_2) \in E_2, \dots, \Phi(M_n) \in E_n, \quad (2.4)$$

где  $\Lambda$  - множество вариантов построения модели  $B_0$ ;  $M, M_1, M_2, \dots, M_n$  - вспомогательные показатели;  $\Phi$  - область значений функций;  $E, E_1, E_2, E_3$  - области допустимых значений для показателей эффективности.

Процесс синтеза имеет структуру:

$$S: X \rightarrow R \rightarrow O \rightarrow Y; \quad (2.5)$$

где  $X$  - исходные данные;  $R$  - результаты преобразования и связывания данных;  $O$  - синтезированные модели;  $Y$  - преобразованные модели.

Возможен двунаправленный процесс синтеза. Тогда определяются переходы вида:

$$\tilde{S}: Y \rightarrow X. \quad (2.6)$$

При этом модель  $X$  рассматривается в качестве целевой модели.

## 2.3. Система показателей и критериев эффективности многоуровневого синтеза моделей объектов мониторинга

Для оценки эффективности многоуровневого синтеза предложена система показателей и критериев эффективности (рисунок 2.3) [138], [484]. Она построена в соответствии с рассмотренным процессом многоуровневого синтеза. На нижнем уровне определяются показатели эффективности обработки контента информационных потоков, поступающих от



наблюдаемых объектов, с учетом контекста. К ним относятся степень адаптивности обработки, информативность формируемых результатов, объем обрабатываемых данных. Адаптивность определяется способностью сохранять значения оценок результатов трансформации данных при изменении контента и контекста. Показатели эффективности обработки данных и способы оценки результатов обработки рассмотрены в работах автора [163].

На основе полученных результатов обработки данных строятся модели наблюдаемых объектов. Для их описания используются автоматные модели. При оценке автоматных моделей рассматриваются полнота синтезируемых моделей и вычислительная сложность их синтеза. Критерии эффективности автоматных моделей сформулированы в [164], [165]. Также для оценки синтезируемых моделей применимы критерии точности, достоверности, адекватности и другие, определяемые в теории математического моделирования.

Эффективность синтезируемых процессов мониторинга определяется временем достижения поставленных целей и объемом затрат, необходимых для перестройки процессов при изменении условий мониторинга.

Для оценки синтезируемых программ мониторинга применяются следующие показатели: объем временных и вычислительных ресурсов, затрачиваемых на построение программ, длина программ. Могут использоваться другие показатели эффективности, применяемые для оценки программ [166].

На верхнем уровне определяются показатели эффективности решения прикладных задач, связанных с мониторингом состояния наблюдаемых объектов. К ним относятся число одновременно наблюдаемых объектов, число успешно решаемых задач, точность и достоверность формируемых результатов, время, затрачиваемое на их получение, объем расходуемых ресурсов и другие.

Содержательная оценка синтезируемых моделей наблюдаемых объектов может строиться на основе анализа их графовых представлений. При этом учитываются типы связей, их полнота и плотность, глубина связанности, функциональная и структурная сложность моделей. Применимы также другие метрики, предлагаемые в теории графов [167], [168].

Оценки эффективности многоуровневого синтеза даются на основе сравнения эффективности решения прикладных задач с применением синтезируемых моделей относительно эффективности их решения с использованием традиционных моделей.



Рисунок 2.3 – Система показателей и критериев эффективности многоуровневого синтеза

К ключевым показателям многоуровневого синтеза автоматных моделей относятся полнота синтезируемых моделей и вычислительная сложность их синтеза [484]. Полнота определяется соотношением известных и не известных параметров модели:

$$Q_r = 1 - \frac{1}{k} \sum_{j=1}^k q_{jr}, \quad (2.7)$$

$$\text{где } q_{1r} = \frac{\|DA(M_{r-1})|-Z_1\|}{Z_1}; \quad q_{2r} = \frac{\|DB(M_{r-1})|-Z_2\|}{Z_2}; \quad q_{3r} = \frac{\|DC(M_{r-1})|-Z_3\|}{Z_3};$$

$$q_{4r} = \frac{\|FB(M_{r-1})|-Z_4\|}{Z_4}; \quad q_{5r} = \frac{\|FC(M_{r-1})|-Z_5\|}{Z_5}, \quad M_{r-1} - \text{модель на } (r-1) \text{ момент времени;}$$

$|DA|, |DB|, |DC|$  - мощности допустимых множеств входных, внутренних и выходных элементов;  $|FB|, |FC|$  - мощности допустимых множеств функций переходов и выходов автомата;  $Z_1, Z_2, \dots, Z_5$  - мощности соответствующих множеств;  $k, k=5$  - коэффициент, назначаемый в соответствии с числом параметров модели. При  $Q_r=1$  модель является полной, при  $Q_r=0$  параметры модели не определены.

Вычислительная сложность синтеза определяется числом операций или временем, необходимым для построения модели. Верхняя граница времени  $T_H$  определяется как

$$T_H \approx c \sum_{i=1}^K m_i^2 \leq c \left( \sum_{i=1}^K m_i \right)^2, \quad (2.8)$$

где  $c$  – постоянный коэффициент;  $m_i$  – число условий задачи на  $i$ -м уровне. Эта оценка справедлива, когда число условий задач многоуровневого и одноуровневого синтеза одинаково. Принимая во внимание, что на каждом верхнем уровне один шаг синтеза эквивалентен  $n_i$  шагам на уровне “0”, оценка нижней границы времени  $T_L$  при многоуровневом синтезе равна:

$$T_L \approx c \sum_{i=0}^K \frac{m_i^2}{n_i^2} \leq c \sum_{i=0}^K m_i^2, \quad (2.9)$$

Тогда среднее время вычисляется как  $T = (T_L + T_H) / 2$ . В случаях, когда при синтезе моделей объектов известны вероятности перехода  $p_i$  на более низкие уровни, для оценки сложности используется следующее выражение:

$$T \approx cn_0^2 + c \sum_{i=0}^K p_i \frac{m_i^2}{n_i^2}. \quad (2.10)$$

## 2.4. Многоуровневые автоматные модели объектов мониторинга

Построение моделей объектов мониторинга осуществляется в дискретном пространстве их состояний и дискретном времени. Тогда каждую из этих моделей можно рассматривать как формальный автомат, характеризующийся своими текущими внутренними, входными и выходными состояниями, а также функциями переходов и выходов, на которые могут накладываться свои ограничения.

Для формального описания моделей в дискретном пространстве и дискретном времени предложено использовать относительно конечные операционные автоматы (ОКА) [169]–[170]. Это класс автоматов, у которых множества допустимых параметров, в общем случае, конечны только на интервале одного шага поведения. При этом имеется возможность изменять в автомате множества допустимых входных, внутренних и выходных состояний, а также множества допустимых функций переходов и выходов автомата, т.е. полностью перестраивать автомат.

### 2.4.1. Многоуровневый относительно конечный автомат

Каждый автомат  $OKA_r$  на  $r$ -ом шаге в полном виде задается совокупностью десяти параметров:

$$OKA_r = \{d_{a_r}, d_{b_r}, d_{c_r}, F_r^b, F_r^c, DA(d_{b_{r-1}}), DB(d_{b_{r-1}}), DC(d_{b_{r-1}}), FB(d_{b_{r-1}}), FC(d_{b_{r-1}})\}; \quad (2.11)$$

где  $d_{a_r}$  - вектор входных данных;  $d_{b_r}$  - вектор параметров внутреннего состояния;  $d_{c_r}$  - вектор параметров состояния выхода. Входящие в (2.11) функции  $F_r^b$  переходов автомата из одного внутреннего состояния в другое и функции  $F_r^c$  выходов можно записать в виде

$$d_{b_{r+1}} = F_r^b(d_{a_r}, d_{b_r}), d_{c_r} = F_r^c(d_{a_r}, d_{b_r}). \quad (2.12 - 2.13)$$

Состояния  $d_{b_r}$ ,  $d_{c_r}$ ,  $d_{a_r}$ , и функции  $F_r^b$ ,  $F_r^c$ , характеризующие автомат на  $r$ -й момент времени, должны удовлетворять условиям:

$$d_{a_r} \in DA(d_{b_{r-1}}) \quad d_{b_r} \in DB(d_{b_{r-1}}), \quad d_{c_r} \in DC(d_{b_{r-1}}), \quad (2.14 - 2.16)$$

$$F_r^b \in FB(d_{b_{r-1}}), \quad F_r^c \in FC(d_{b_{r-1}}). \quad (2.17 - 2.18)$$

Условие (2.14) отражает, что состояние входов автомата на  $r$ -й момент времени ограничивается множеством  $DA(d_{b_{r-1}})$  допустимых состояний, определенных относительно  $r-1$  момента времени. Согласно (2.15) внутреннее состояние автомата на  $r$ -й момент времени должно относиться к множеству  $DB(d_{b_{r-1}})$  его допустимых внутренних состояний. Выражение (2.16) характеризует ограниченность возможных состояний выходов автомата. Эти состояния должны относиться к множеству  $DC(d_{b_{r-1}})$ . По условию (2.17) реализуемая автоматом на  $r$ -й момент функция переходов  $F_r^b$  должна входить в множество  $FB(d_{b_{r-1}})$  допустимых функций, определенное относительно  $r-1$  момента времени. Множество  $FB(d_{b_{r-1}})$  функций переходов отражает систему переходов, характерных автомату на  $r$ -й момент времени.

По условию (2.18) функция  $F_r^c$  выходов на  $r$ -й момент времени должна принадлежать множеству  $FC(d_{b_{r-1}})$  допустимых функций, активных относительно  $r-1$  момента времени.

Переход от автомата  $OKA_r$  к автомату  $OKA_{r+1}$  на  $r+1$  представим в виде

$$F_r^b : OKA_r, d_{a_r} \rightarrow OKA_{r+1}. \quad (2.19)$$

При логическом варианте записи функции переходов автомата (2.12) из одного состояния в другое представляются в виде

$$F_r^b(d_{a_r}, d_{b_r}) \rightarrow d_{b_{r+1}}. \quad (2.20)$$

Если расширить множество внутренних состояний не только на состояния выхода, но и входа, то в (2.20) зависимость  $F_r^b(\cdot)$  от  $d_{a_r}$  можно не показывать.

В случае, если при переходах используются многие функции, то переходы описываются следующим образом:

$$F_{zv}(d_{z^v_e}; e = \overline{1, E_z}) \rightarrow d_{z^v_a}; z = \overline{1, Z}; v = \overline{1, V_z}, \quad (2.21)$$

где  $z$  и  $v$  определяют вид и тип функций, которые используются при переходах.

Для каждого из условий может быть определен статус: основное условие, предусловие, постусловие. Для предусловий указываются номера основных условий (ОУ), при которых они истинны. Постусловиям ставятся в соответствие номера ОУ, при которых они ложны. Тогда всем элементам в условиях могут быть сопоставлены предикаты,  $P_{zv0}(F_{zv0}(\bullet))$ ,  $P_{zv1}(d_{zv1}), \dots, P_{zva}(d_{zva})$ , принимающие значение 1, когда переменные определены (истинны), и 0 в противном случае. В этом случае условия представляют собой структуры вида

$$\{P_{zv0} \wedge P_{zv1} \wedge \dots \wedge P_{zve_z} \rightarrow P_{zva} \mid S_{zv}, M_{zv}, z = \overline{1, Z}; v = \overline{1, V}\}, \quad (2.22)$$

где  $S_{zv}$  - статус  $zv$ -го условия,  $M_{zv}$  - множество номеров ОУ, при которых предусловия истинны, а постусловия ложны.

С учетом рассмотренных соотношений автоматная модель наблюдаемого объекта описывается в виде структуры:

$$OKA = \{F_{zv}(d_{zve}; e = \overline{1, E_z}) \rightarrow d_{zva}; \{d_s\}; \{d_w\}; z = \overline{1, Z}; v = \overline{1, V_z}\}, \quad (2.23)$$

связывающей  $\{d_s\}$  и  $\{d_w\}$  между собой.

На основе одноуровневых автоматов автором разработаны иерархические относительно-конечные операционные автоматы [164], [165]. При этом сформулирована и решена новая задача построения автоматных моделей с иерархической структурой.

Для построения иерархических ОКА выделены опорные множества допустимых параметров в таких автоматах. Эти множества отмечаются верхним индексом «0», при этом опускается их зависимость от внутренних состояний. Совокупность таких опорных множеств обозначается как:

$$DOKA^0 = \{DA^0, DB^0, DC^0, FB^0, FC^0\}. \quad (2.24)$$

Из элементов этих опорных множеств (2.24) сформированы допустимые множества параметров более высоких  $i$ -х уровней,  $DOKA^i = \{DA^i, DB^i, DC^i, FB^i, FC^i\}$ . В результате автомат может характеризоваться допустимыми множествами параметров на различных уровнях иерархии:

$$DOKA^0 \Leftrightarrow DOKA^1 \Leftrightarrow \dots \Leftrightarrow DOKA^i \Leftrightarrow \dots \Leftrightarrow DOKA^K. \quad (2.25)$$

Учитывая, что автомат на  $i$ -ом шаге описывается  $DOKA^i = \{DA^i, DB^i, DC^i, FB^i, FC^i\}$  и эти множества, в общем случае, изменяются во времени, то их можно записывать как зависящие от его внутренних состояний,  $DOKA^i = DOKA^i(\bar{d}_{b_{r-1}})$ .

В развернутом виде допустимые множества представляются следующим образом:

$$\left\{ \begin{array}{l} DA^i(\bar{d}_{r-1}) \xrightarrow{F_{ij}^{DA}} DA^j(\bar{d}_r) \\ DB^i(\bar{d}_{r-1}) \xrightarrow{F_{ij}^{DB}} DB^j(\bar{d}_r) \\ DC^i(\bar{d}_{r-1}) \xrightarrow{F_{ij}^{DC}} DC^j(\bar{d}_r) \\ FB^i(\bar{d}_{r-1}) \xrightarrow{F_{ij}^{FB}} FB^j(\bar{d}_r) \\ FC^i(\bar{d}_{r-1}) \xrightarrow{F_{ij}^{FC}} FC^j(\bar{d}_r) \end{array} \right\}, \quad (2.26)$$

где  $F_{ij}^{DA}, F_{ij}^{DB}, F_{ij}^{DC}, F_{ij}^{FB}, F_{ij}^{FC}$  - функции, применяемые для построения множеств допустимых параметров.

В случаях, если множество внутренних состояний иерархического автомата расширяется на состояния его выхода, то строится усеченный по параметрам автомат, но с сохранением свойств по перестройке параметров:

$$DOKA_r = \{d_{a_r}, d_{b_r}, F_r^b, DA(d_{b_{r-1}}), DB(d_{b_{r-1}}), FB(d_{b_{r-1}})\}. \quad (2.27)$$

Таким образом, иерархические ОКА позволяют рассматривать объекты в виде совокупности взаимосвязанных элементов более низких уровней. Переход от одного уровня к другому с формальной точки зрения сводится к перестройке множеств допустимых параметров автоматной модели. При этом на более высоких уровнях число элементов может быть значительно меньше, чем на нулевом опорном уровне.

Функции переходов для иерархических ОКА описываются в виде:

$$\left\{ \begin{array}{l} \{d_s^0\} \Rightarrow \{F_{zv}^0(d_{zv_e}^0; e = \overline{1, E_z}) \rightarrow d_{zv_a}^0; z = \overline{1, Z_i}; v = \overline{1, V_{iz}}\}; \Rightarrow \{d_w^0\} \\ \{d_s^1\} \Rightarrow \{F_{zv}^1(d_{zv_e}^1; e = \overline{1, E_z}) \rightarrow d_{zv_a}^1; z = \overline{1, Z_i}; v = \overline{1, V_{iz}}\}; \Rightarrow \{d_w^1\} \\ \dots \\ \{d_s^k\} \Rightarrow \{F_{zv}^k(d_{zv_e}^k; e = \overline{1, E_z}) \rightarrow d_{zv_a}^k; z = \overline{1, Z_i}; v = \overline{1, V_{iz}}\}; \Rightarrow \{d_w^k\} \\ \dots \\ \{d_s^N\} \Rightarrow \{F_{zv}^N(d_{zv_e}^N; e = \overline{1, E_z}) \rightarrow d_{zv_a}^N; z = \overline{1, Z_i}; v = \overline{1, V_{iz}}\}; \Rightarrow \{d_w^N\} \end{array} \right\}; \quad (2.28)$$

При этом при переходах перестраиваются допустимые множества параметров на каждом из уровней.

Для предложенных относительно-конечных автоматных моделей предусмотрена возможность определения вложенных автоматов. В результате обеспечивается возможность формирования многоуровневых систем вложенных автоматных моделей. Многоуровневые вложенные автоматные модели рассмотрены в статье автора [173]. Применение вложенных моделей позволяет выполнять локальную перестройку моделей при изменении условий мониторинга.

## 2.4.2. Функциональное пространство и пространство состояний многоуровневого относительно конечного автомата

Для описания функциональных возможностей синтезируемых автоматных моделей предусмотрены функциональные модели [162] (рисунок 2.4). В них функции разнесены по уровням и связаны между собой. Функциональные модели описываются в виде:

$$\{FB^i, FC^i\}_r \rightarrow \{FA_j^i\}_r, \quad (2.29)$$

где  $FA_j^i$  -  $j$ -ая функция автоматной модели на уровне  $i$ . Функции определяются условиями, наблюдаемыми на шаге  $r$  и допустимыми множествами функций внутренних переходов и выходов.

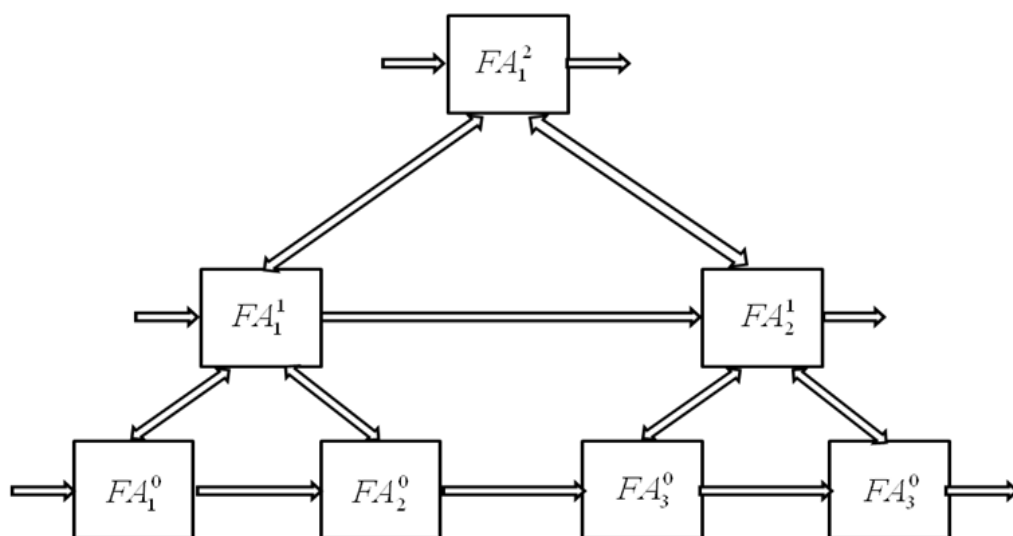


Рисунок 2.4 – Иерархическая структура функций иерархических ОКА

Пространство состояний рассматриваемого автомата также имеет иерархическую структуру.

Пространства состояний и функциональные пространства, которые построены для условий одного шага, определяют автоматы, являющиеся локальными моделями объектов. Если при построении автоматов рассматриваются различные условия, то результирующие автоматы представляют собой глобальные модели объектов.

При формировании пространств состояний и функциональных пространств определяются допустимые функции и допустимые состояния. Для этого осуществляется обработка исходных данных. Она состоит в обобщении частных примеров, содержащихся в полученных от объектов данных.

При обработке данные рассматриваются на отдельных временных интервалах, на которых состояние объекта не изменялось. Для каждого такого интервала времени строится многоуровневая структура, описывающая наблюдаемый объект на этом интервале. В результате формируются последовательности связанных между собой структур.

Формально на каждом  $z$ -ом уровне структура определяется как

$$K_z = K_z(S_z(e_{zi}, l_{zij}), \{e_{zi}\}, \{l_{zij}\}; e_{zi} \in E_z; l_{zij} \in L_z), \quad (2.30)$$

где  $S_z(e_{zi}, l_{zij})$  - структура уровня  $z$ ,  $\{e_{zi}\}$  и  $\{l_{zij}\}$  - информационные элементы и связи, содержащиеся в структуре,  $E_z$ ,  $L_z$  - множества допустимых элементов и связей между элементами на  $z$  - ом уровне.

Между элементами, относящимися к различным уровням, устанавливаются вертикальные связи  $h_{zrj} \in H_{zr}$ , где  $H_{zr}$  - множество допустимых вертикальных  $zr$  связей.

Структуры  $K_z$  ассоциируются с интервалами времени, для которых они определены, или с отдельными моментами времени из этих интервалов. Такие структуры являются статическими структурами, которые характеризуют объекты на рассматриваемых интервалах. Для описания объектов на более длительных временных интервалах построенные статические структуры связываются между собой, формируя при этом динамические структуры. Динамическая структура - это единая скользящая во времени структура, которая определена на временном интервале  $[t_0, t_N]$ :

$$G = G(S(K_r, d_{rj}), \{K_r\}, \{d_{rj}\}); r \in [t_0, t_N]; d_{rj} \in D_r, \quad (2.31)$$

где  $\{d_{rj}\}$  - информационные связи динамической структуры, построенной на шаге  $r$ ;  $D_r$  - множества допустимых связей между элементами статических структур на шаге  $r$ .

На рисунке 2.5. показан пример построения структуры, характеризующей объект на временном интервале. Представленный на рисунке объект включает три составляющих, для каждой из которых строится своя структура. При этом между отдельными информационными элементами устанавливаются горизонтальные  $F_h$  и вертикальные  $F_v$  связи. За счет горизонтально-вертикального связывания  $F_{hv}$  структур, построенных для составляющих объекта, формируется структура, характеризующая объект в целом. Для построения динамических структур на основе статических устанавливаются связи  $F_t$ , которые отражают изменения статических структур во времени.

Заметим, что такие структуры могут быть представлены в виде относительно конечных автоматов.

Подробное описание статических и динамических структур можно найти в [161], вопросы их построения рассмотрены в [174], [175].



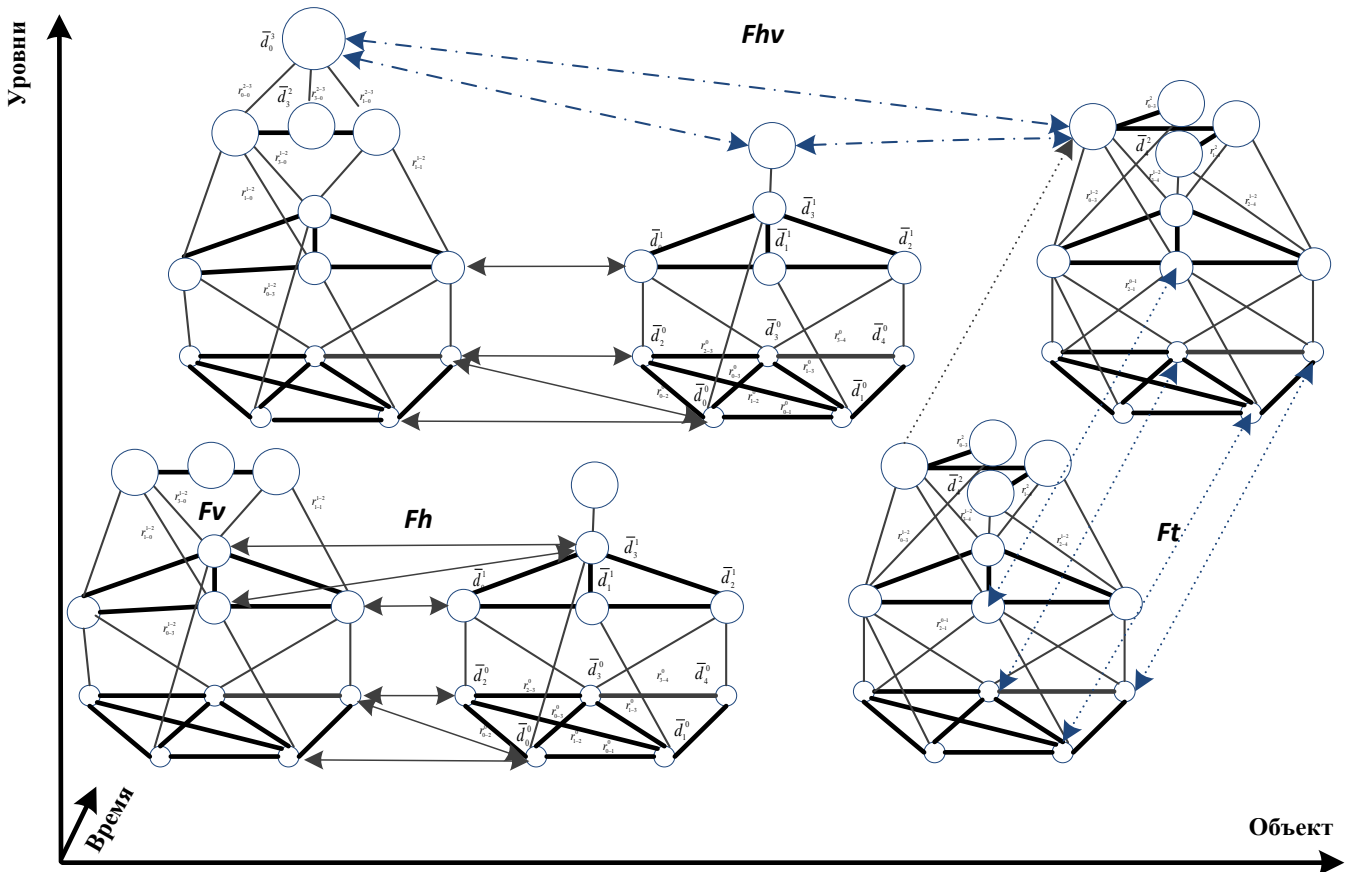


Рисунок 2.5 – Статические и динамические структуры, характеризующие наблюдаемые объекты

## 2.5. Типовые формулировки задач многоуровневого синтеза моделей объектов мониторинга

Для синтеза моделей объектов в каждом конкретном случае формулируются и решаются частные задачи. В соответствии с уровнями системы показателей и критериев эффективности выделяются следующие группы частных задач: прикладные задачи, задачи синтеза моделей процессов и программ мониторинга, задачи синтеза моделей объектов мониторинга, задачи трансформации контента в контексте [138], [484]. Ниже приводятся отдельные частные задачи.

К типовым задачам прикладного уровня относится задача поиска варианта построения моделей объектов, процессов и программ мониторинга  $V_o$ , обеспечивающего достижение максимума числа одновременно наблюдаемых объектов с периодичностью  $P$  не ниже заданной:

$$V_o(t) \in \underset{i \in \Omega}{\text{Arg max}} Q(V_i(K(t)), K(t), C(t), t),$$

$$P(V_i(K(t)), K(t), C(t), t) \geq P_{\text{зад}},$$

где  $V_i(K(t))$  - варианты построения моделей объектов, процессов и программ мониторинга;  $C(t)$  - контент;  $K(t)$  - контекст,  $\Omega$  - множество вариантов построения моделей объектов, процессов и программ мониторинга.

Другой типовой задачей является задача поиска варианта синтеза моделей объектов, процессов и программ мониторинга  $V_0$ , обеспечивающего достижение максимума информативности  $G$  синтезируемых моделей объектов при ограничениях на точность  $Z$  и достоверность  $R$ :

$$V_0(t) \in \text{Arg max}_{i \in \Omega} G(Z(V_i, R), R(V_i), K(t), C(t), t)$$

$$Z(V_i(K(t)), R(V_i), t) \geq Z_{\text{зад}}; R(V_i(K(t)), K(t), C(t), t) \geq R_{\text{зад}}.$$

Возможна постановка задачи поиска варианта синтеза моделей объектов, процессов и программ  $V_0$ , обеспечивающего получение необходимой информации за минимальное время  $T_0$  при заданных допустимых значениях точности  $Z$  и информативности  $G$  синтезируемых моделей:

$$V_0(t) \in \text{Arg min}_{i \in \Omega} T(V_i(K(t)), K(t), C(t), t);$$

$$G(Z(V_i), R(V_i), K(t), t) \geq G_{\text{дон}};$$

$$Z(V_i(K(t)), R(V_i), t) \geq Z_{\text{дон}}; R(V_i(K(t)), K(t), C(t), t) \geq R_{\text{дон}}.$$

Поиск варианта синтеза моделей объектов, процессов и программ  $V_0$  может также выполняться по критерию минимума расхода ресурсов, обеспечивающего заданную информативность результирующей модели:

$$V_0(t) \in \text{Arg min}_{i \in \Omega} U(V_i(K(t)), K(t), C(t), t);$$

$$G(Z(V_i), R(V_i), K(t), t) \geq G_{\text{зад}}; Z(V_i(K(t)), R(V_i), t) \geq Z_{\text{дон}}; R(V_i(K(t)), K(t), C(t), t) \geq R_{\text{дон}}.$$

*Частные задачи синтеза процессов и программ мониторинга.* При синтезе процессов и программ мониторинга можно исходить из достижения максимальной скорости обработки информации  $D$  при ограничениях на точность обработки  $Z$ , которая определяется объемом информации  $I$  и видами обработки  $L$ :

$$V_0(\Delta t) \in \text{Arg max}_{i \in \Omega} D(V_i(K(\Delta t)), K(\Delta t));$$

$$Z(V_i, I(\Delta t), L(\Delta t)) \geq Z_{\text{зад}}.$$

Синтез процессов и программ мониторинга  $V_0$  можно выполнять, требуя минимизацию затрат на перестройку существующих процессов  $U$  с учетом ожидаемого повышения информативности моделей объектов:

$$V_0(t) \in \underset{i \in \Omega}{\text{Arg min}} U(\Delta V(t), \Delta G(t));$$

$$\Delta V(t) = |V_i(t) - V'|; \Delta G(t) = |G(V_i(t)) - G(V')|.$$

При синтезе процессов мониторинга может обеспечиваться максимальная информативность результатов обработки входного контента при ограничениях на ресурсы:

$$V_0(\Delta t) \in \underset{i \in \Omega}{\text{Arg max}} G(Z(V_i), R(V_i), K(\Delta t));$$

$$U(V_i(K(\Delta t))) \leq U_{\text{зад}}.$$

*Частные задачи синтеза моделей объектов мониторинга.* При синтезе моделей объектов может ставится задача поиска варианта синтеза  $V_0$ , обеспечивающего построение наиболее полной модели по имеющимся контенту и контексту с учетом возможностей получения дополнительной информации при ограничениях на достоверность  $R$  и точность  $Z$  модели, а также ресурсы  $U$  для ее построения:

$$V_0(t) \in \underset{i \in \Omega}{\text{Arg max}} G(Z(V_i), R(V_i), K(t), \Delta K);$$

$$R(V_i, \Delta K) \geq R_{\text{дон}}; Z(R(V_i), \Delta K) \geq Z_{\text{зад}}; U(V_i(\Delta K)) \leq U_{\text{зад}}.$$

Другой возможной задачей синтеза моделей объектов мониторинга является поиск варианта  $V_0$ , обеспечивающего максимум достоверности моделей мониторинга с учетом возможностей получения дополнительной информации при ограничениях на полноту  $G$  и затрачиваемое время  $T$ . При синтезе оценивается достоверность каждого из  $j$ -ых элементов модели:

$$V_0(t) \in \underset{i \in \Omega}{\text{Arg max}} R(V_i(K(t)), K(t), \Delta K);$$

$$R(V_i) = (1 - \prod_j (1 - R_j(V_i))); G(Z(V_i), R(V_i), \Delta K) \geq G_{\text{зад}}; T(V_i(\Delta K)) \leq T_{\text{зад}}.$$

*Частные задачи трансформации данных мониторинга.* При решении задач трансформации данных может требоваться определить такой вариант  $V_0$  трансформации, который обеспечит достижение минимума времени, необходимого для обработки данных, при условии, что точность  $Z$  преобразования и достоверность  $R$  результата будет не ниже требуемой:

$$V_0(t) \in \underset{i \in \Omega}{\text{Arg min}} T(V_i(K(t)), C(t), t),$$

$$Z(R(V_i), t) \geq Z_{\text{зад}}; R(V_i, K(t), C(t), t) \geq R_{\text{дон}}.$$

В другой постановке задачи трансформации может потребоваться определить вариант трансформации  $V_0$ , обеспечивающий максимум точности преобразований при ограничениях на отводимые ресурсы  $U$ :

$$V_0(t) \in \text{Arg max}_{i \in \Omega} Z(R(V_i), K(t), C(t), t);$$

$$R(V_i(K(t)), K(t), C(t), t) \geq R_{\text{дон}}; U(V_i(K(t)), K(t), C(t), t) \leq U_{\text{зад}}.$$

Также при решении задач трансформации данных может выполняться поиск варианта трансформации  $V_0$ , обеспечивающего максимум информативности контента на единицу объема обрабатываемого потока  $K(t)_{\Delta_1}$  при ограничениях на время обработки:

$$V_0(t) \in \text{Arg max}_{i \in \Omega} G(Z(V_i), R(V_i), K_{\Delta_1}(t), K(t), C(t), t);$$

$$T(V_i(K_{\Delta_1}(t)), K_{\Delta_1}, K(t), C(t), t) \leq T_{\text{зад}}.$$

## 2.6. Обобщенный алгоритм многоуровневого синтеза моделей объектов мониторинга

Для решения задач многоуровневого синтеза предложен обобщенный алгоритм, позволяющий синтезировать модели объектов в соответствии с поставленными целями синтеза и заданными критериями эффективности. Этот алгоритм позволяет целенаправленно исследовать объект мониторинга с целью восстановления его модели, т.е. определения всех параметров ОКА, описывающих этот объект. Подробное описание алгоритма приведено в работе [10], [11]. Общая структура алгоритма показана на рисунке 2.6.

Согласно алгоритму вначале формулируются частные задачи синтеза моделей объектов мониторинга в соответствии с требованиями, предъявляемыми ожидаемыми потребителями моделей. Модели могут применяться для идентификации, классификации, распознавания, прогнозирования изменения состояний наблюдаемых объектов, управления ими и решения других задач.

Для построения моделей объектов выполняется контентно-адаптивная обработка информационных потоков, получаемых от наблюдаемых объектов и из других источников. При обработке определяются характеристики контента потоков, выявляются их особенности. Контекст обработки определяется состоянием среды, в которой функционируют объекты, условиями выполнения обработки, возможностями применяемых систем сбора данных. Значимую роль при обработке может иметь априорная информация об объектах и накопленные статистические данные.

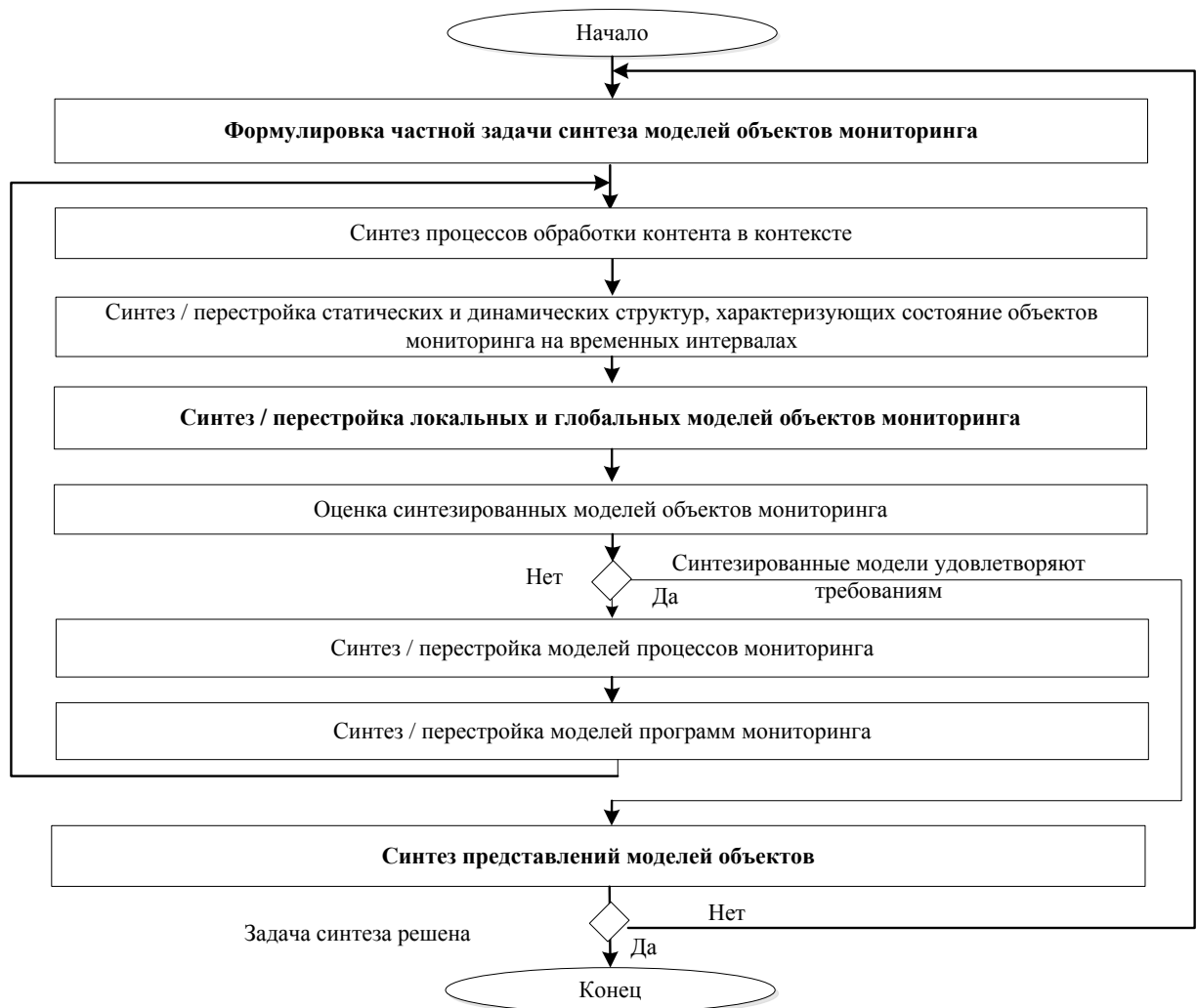


Рисунок 2.6 – Обобщенный алгоритм многоуровневого синтеза моделей объектов мониторинга

По результатам обработки данных строятся модели объектов мониторинга или перестраиваются существующие. При высокой динамике изменения состояния среды, наличия у объектов сложного поведения, перестройка моделей объектов выполняется достаточно часто. Для построенных моделей определяются их параметры, выполняется оценка моделей. Выявляются несоответствия между моделями, отражающими фактическое состояние объектов, которое определено на основе информации, извлеченной из контента, и моделями, необходимыми для решения практических задач. В случае, если синтезированные модели не соответствуют требуемым, то определяются данные, необходимые для перестройки имеющихся моделей. Для сбора данных синтезируются процессы мониторинга и соответствующие им программы мониторинга. Цикл, включающий перестройку моделей объектов, процессов и программ выполняется, пока не будут построены требуемые модели или не будет доказано, что их построение не возможно.

При успешном синтезе моделей формируются их представления. Представления содержат часть данных, содержащихся в моделях, которые необходимы для решения потребителями их задач.

Задача синтеза моделей объектов мониторинга считается решенной, если результирующие модели обеспечили решение прикладных задач. При этом полученные решения удовлетворяют критериям эффективности.

## 2.7. Выводы

1. Разработаны концептуальные положения многоуровневого автоматического синтеза моделей объектов мониторинга. Предложена концептуальная модель синтеза моделей объектов по получаемым от них данным. Объекты описываются через взаимосвязи между их элементами. Для сбора необходимых данных синтезируются процессы мониторинга и соответствующие им формальные программы, которые могут быть описаны на скриптовых языках. Модель имеет следующие отличительные особенности: синтез является многоуровневым, а синтезируемые модели - формальными полностью перестраиваемыми моделями. Синтез на многих уровнях решает проблему вычислительной сложности. Применение перестраиваемых автоматных моделей при описании моделей объектов мониторинга обеспечивает широкие возможности по их автоматической перестройке.

2. Предложена машина многоуровневого синтеза моделей объектов мониторинга, представляющая собой абстрактный вычислитель, обеспечивающий решение задач синтеза моделей объектов мониторинга. Она предусматривает построение моделей контента и моделей контекста, отражающих содержание исходных данных и условия синтеза моделей. На их основе строятся модели объектов. Для перехода от построенных моделей к требуемым синтезируются процессы мониторинга. Они позволяют получать данные, необходимые для построения требуемых моделей на основе имеющихся.

3. Предложена система взаимосвязанных показателей и критериев эффективности многоуровневого синтеза моделей объектов мониторинга. Система показателей включает показатели эффективности обработки контента информационных потоков, поступающих от наблюдаемых объектов, показатели эффективности синтеза моделей объектов мониторинга, а также процессов и программ мониторинга. На верхнем уровне определяются показатели эффективности решения прикладных задач, среди которых число одновременно наблюдаемых объектов, число успешно решаемых задач, точность и достоверность формируемых результатов, время, затрачиваемое на их получение, объем расходуемых ресурсов. Ключевыми показателями многоуровневого синтеза автоматных моделей выступают полнота синтезируемых моделей и вычислительная сложность их синтеза.

3. Разработаны модели объектов мониторинга. Модели объектов предложено описывать в дискретных пространствах их состояний в виде относительно конечных операционных автоматов. Это класс автоматов, способных к полной перестройке. Построена новая модель

иерархических ОКА, развивающая существующие одноуровневые модели автоматов. Раскрыто функциональное пространство и пространство состояний предложенных иерархических ОКА.

4. Сформулированы общая и частные задачи многоуровневого синтеза моделей объектов мониторинга. Группы частных задач определены в соответствии с системой показателей и критериев эффективности. Они включают группу прикладных задач, задач синтеза моделей процессов и программ мониторинга, задач синтеза моделей объектов мониторинга, задач трансформации контента в контексте.

5. Предложен обобщенный алгоритм многоуровневого синтеза моделей объектов мониторинга. Алгоритм определяет основные шаги, выполнение которых позволяет синтезировать модели объектов в соответствии с поставленными целями синтеза и заданными критериями эффективности.

В целом, полученные в главе два результаты формируют основы теории многоуровневого автоматического синтеза автоматных моделей объектов мониторинга.

## Глава 3. Методы многоуровневого синтеза моделей объектов, процессов и программ мониторинга

### 3.1. Методы многоуровневого синтеза автоматных моделей объектов мониторинга

#### 3.1.1. Постановка задачи синтеза автоматных моделей объектов мониторинга

Задача автоматического синтеза моделей объектов состоит в синтезе многоуровневых автоматных моделей, описывающих состояния этих объектов на временных интервалах. Такой синтез осуществим с применением индуктивного или дедуктивного подходов. Согласно индуктивному подходу синтез осуществляется путем анализа и обобщения частных примеров, переходом от частного к общему. При дедуктивном синтезе следуют от общего к частному: формулируется теорема о существовании модели на заданных условиях, доказывается ее существование и из доказательства извлекается сама интересующая модель. Однако применение традиционных методов одноуровневого автоматического синтеза сталкивается с высокой сложностью решаемых задач, что существенно ограничивает использование их на практике. Существующие модели и методы синтеза позволяют описывать одноуровневые структуры вида (2.23).

*Постановка задачи многоуровневого синтеза автоматных моделей объектов.* Формулировка задачи предусматривает многоуровневый синтез автоматных моделей объектов мониторинга на основе получаемых от них данных. Согласно этой формулировке известны исходные данные  $\{d_s(t_i); t_i \in [1, T]\}$  на  $t_i$  моменты времени из некоторого интервала времени  $[1, T]$ . В общем случае требуется построить многоуровневую автоматную модель на основе этих данных, определенную на интересующем временном интервале:  $DOKA(t_k); t_k \in [1, T]$ . Накладываются условия вида

$$\{d_s(t_k)\} \rightarrow \left\{ \left\{ d_s^0(t_k) \right\}, \left\{ d_k^1(t_i) \right\}, \dots, \left\{ d_k^j(t_i) \right\}, \dots, \left\{ d_k^L(t_i) \right\} \right\}; t_k \in [0; T] \}. \quad (3.1)$$

В соответствии с ними на каждый момент времени данные должны быть представимы по уровням, число уровней  $L$  конечно.

Для перехода к постановке задачи синтеза моделей, характеризующих объекты на фиксированные моменты времени или на временных интервалах, на которых состояние объектов не изменяется, достаточно исключить из рассмотрения параметр времени  $t$ .

#### 3.1.2. Метод многоуровневого синтеза автоматных моделей объектов мониторинга

Для решения задачи многоуровневого синтеза моделей объектов предложен новый метод [162], предусматривающий совместное использование индуктивного и дедуктивного синтеза [174], [173]. В состав метода входят следующие шаги:



1. Формирование условно стационарных интервалов  $T = \sum_{j=0}^{N-1} T_j$ .

2. Многоуровневый индуктивный синтез статических и динамических структур  $d_w^i : F_{zv}^i(d_{zv_e}^i; e = \overline{1, E_z}) \rightarrow d_{zv_a}^j; z = \overline{1, Z_i}; v = \overline{1, V_{iz}}; i, j = \overline{1, L}$ ; на основе обработки исходных данных.

3. Многоуровневый синтез локальных моделей объектов. При этом на основе синтезированных структур для каждой модели строятся допустимые множества элементов и функций их связывающих  $\{DA^i, DB^i, DC^i, FB^i, FC^i\}$ .

4. Многоуровневый синтез глобальных моделей объектов. При этом допустимые множества состояний определяются на основе локальных моделей объектов. Для построения допустимых множеств функций устанавливаются связи между элементами локальных моделей

вида  $F_{zv_e}^i : (d_{zv_e}^i(t_j), F_{zv_e}^i(t_k)) \xrightarrow{P_{zv_e}^{i,j}(F_{zv_e}^{i,j}(\cdot)), P_{zv_e}^{i,j}(d_{zv_e}^{i,j}(\cdot))} d_{zv_a}^j(t_{k+\Delta t}); z = \overline{1, Z}; v = \overline{1, V_z}; i, j = \overline{1, L}$ ,

$t_k, t_{k+\Delta} \in [0; T]$ ; которые затем обобщаются. В результате формируются требуемые множества  $\{DA^i, DB^i, DC^i, FB^i, FC^i\}$ .

### 3.1.3. Метод индуктивного многоуровневого синтеза автоматных моделей объектов мониторинга

*Постановка задачи многоуровневого индуктивного синтеза моделей объектов.* Задача многоуровневого индуктивного синтеза предусматривает синтез локальных моделей. Заданы исходные данные  $\{d_s(t_i); t_i \in [1, T]\}$  и типы правил их связывающие  $DF_{zv}$ ;  $z = \overline{1, Z}; v = \overline{1, V_z}$ . Требуется по исходным данным построить локальные модели объектов в виде многоуровневых автоматов:

$$F_{zv}^i(d_{zv_e}^i; e = \overline{1, E_z}) \rightarrow d_{zv_a}^i; z = \overline{1, Z_i}; v = \overline{1, V_{iz}}; i = \overline{1, L}; \quad (3.2)$$

$$F_{zv}^i(\cdot) \in FB^i; d_{zv_e}^i, d_{zv_a}^i \in DD^i. \quad (3.3)$$

Для элементов моделей должны быть определены предикаты  $P_{zv_0}(F_{zv_0}(\cdot)), P_{zv_1}(d_{zv_1}), \dots, P_{zv_a}(d_{zv_a})$ , описывающие условия переходов в виде  $\{P_{zv_0} \wedge P_{zv_1} \wedge \dots, P_{ZVE_z} \rightarrow P_{zva} | S_{zv}, M_{zv}\}$ , где  $S_{zv}$  - статус  $zv$ -го условия,  $M_{zv}$  - множество условий, определяющих значения предикатов.

*Метод многоуровневого индуктивного синтеза моделей объектов.* Предлагаемый метод индуктивного синтеза статических моделей предусматривает:

1. Построение стационарных интервалов:  $T_j, \overline{j = 1, N}$ . Построение по исходным данным статических и динамических структур вида (2.31) на каждом из интервалов.

2. Синтез допустимых множеств элементов входных  $DA^0$ , выходных  $DB^0$  данных и внутренних элементов  $DC^0$  модели на нулевом уровне:  $\{DA^0, DB^0, DC^0\}$ .

3. Синтез допустимых множеств связей между элементами  $FB^0$  и с элементами выхода  $FC^0$  модели на нулевом уровне:  $\{FB^0, FC^0\}$ .

4. Синтез допустимых множеств элементов на  $i$  уровне и связей между ними:  $\{DA^i, DB^i, DC^i, FB^i, FC^i\}$ .

5. Синтез расширенного множества допустимых связей  $i$ -го уровня с учетом связей с другими уровнями:  $\{FB^{i,j}, FC^{i,j}; i \neq j; j = \overline{1, L}\}; FB^i \supset FB^{i,j}; FC^i \supset FC^{i,j}$ .

6. Формирование логических условий для элементов и связей между ними:  $P_{zv^0}^i(F_{zv^0}^i(\bullet)), P_{zv^1}^i(d_{zv^1}^i), \dots, P_{zva}^i(d_{zva}^i); \{P_{zv^0}^i \wedge P_{zv^1}^i \wedge, \dots, P_{zve}^i \rightarrow P_{zva}^i \mid S_{zv}^i, M_{zv}^i, z = \overline{1, Z}; v = \overline{1, V}\}$ , где  $S_{zv}^i$  - статус  $zv$  условия;  $M_{zv}^i$  - множество условий, определяющих значения предикатов.

### 3.1.4. Метод дедуктивного синтеза автоматных моделей объектов мониторинга

*Постановка задачи многоуровневого дедуктивного синтеза.* Задача многоуровневого дедуктивного синтеза предусматривает синтез глобальных моделей объектов, отражающих состояние объектов на временных интервалах. Заданы локальные модели, элементы которых разнесены по уровням:

$$\{d_s^0(t_k)\}, \{d_s^1(t_k)\}, \dots, \{d_s^i(t_k)\}, \dots, \{d_s^L(t_k)\}, \quad (3.4)$$

$$\{d_w^0(t_{k+\Delta t})\}, \{d_w^1(t_{k+\Delta t})\}, \dots, \{d_w^i(t_{k+\Delta t})\}, \dots, \{d_w^L(t_{k+\Delta t})\}. \quad (3.5)$$

Требуется за счет связывания локальных моделей между собой построить глобальные модели объектов для интересующего временного интервала с функциями переходов вида:

$$\left\{ \begin{array}{l} F_{zv_e}^0 : (d_{zv_e}^0(t_k), F_{zv_e}^0(t_k)) \xrightarrow{P_{zv_e}^{0,j}(F_{zv_e}^0(\bullet)), P_{zv_e}^{0,j}(d_{zv_e}^0(\bullet))} d_{zva}^j(t_{k \pm \Delta t}) \\ F_{zv_e}^1 : (d_{zv_e}^1(t_k), F_{zv_e}^1(t_k)) \xrightarrow{P_{zv_e}^{1,j}(F_{zv_e}^1(\bullet)), P_{zv_e}^{1,j}(d_{zv_e}^1(\bullet))} d_{zva}^j(t_{k \pm \Delta t}) \\ \dots \\ F_{zv_e}^L : (d_{zv_e}^L(t_k), F_{zv_e}^L(t_k)) \xrightarrow{P_{zv_e}^{L,j}(F_{zv_e}^L(\bullet)), P_{zv_e}^{L,j}(d_{zv_e}^L(\bullet))} d_{zva}^j(t_{k \pm \Delta t}) \end{array} \right. \quad (3.6)$$

При связывании должны учитываться уровни и условия существования связей  $P_{zv}^i$ .

*Метод многоуровневого дедуктивного синтеза моделей объектов.* Предлагаемый метод дедуктивного синтеза глобальных моделей предусматривает:

1. Определение анализируемого интервала  $T = [T_A; T_B]; d_w(t_k) : t_k \in [T_A; T_B]$ .

2. Синтез состава и типов зависимостей, наблюдаемых между элементами локальных моделей на различных уровнях:  $\{F_{zv}^i(t_k, t_{k+\Delta t}); z = \overline{1, Z}; v = \overline{1, V}; i = \overline{1, L}\}$ .

3. Синтез параметров  $S$  установленных зависимостей  $i$  уровня:  $\{F_{zve}^i(S_{zve}, \Delta t); e = \overline{1, E_z}; i = \overline{1, L}\}$ .

4. Синтез расширенного множества связей  $i$ -го уровня с учетом связей с другими  $j$ -ми уровнями:  $\{F_{zve}^{i,j}(S_{zve}, \Delta t)\}; F_{zve}^i \supset F_{zve}^{i,j}$ .

5. Синтез условий, при которых наблюдаются зависимости между элементами:  $\{P_{zve}^{i,j}(F_{zve}^{i,j}(\bullet)), P_{zve}^{i,j}(d_{zve}^{i,j}(\bullet)); i = \overline{1, L}\}$ .

6. Синтез модели вида:

$$\left\{ \begin{array}{l} F_{zve}^0(S_{zve}, \Delta t) | P_{zve}^0 : (d_{zve}^0(t_k), F_{zve}^0(t_k)) \xrightarrow{P_{zve}^{0,j}(F_{zve}^{0,j}(\bullet)), P_{zve}^{0,j}(d_{zve}^{0,j}(\bullet))} d_{zva}^j(t_{k+\Delta t}) | P_{zve}^j \\ \dots \\ F_{zve}^L(S_{zve}, \Delta t) | P_{zve}^L : (d_{zve}^L(t_k), F_{zve}^L(t_k)) \xrightarrow{P_{zve}^{L,j}(F_{zve}^{L,j}(\bullet)), P_{zve}^{L,j}(d_{zve}^{L,j}(\bullet))} d_{zva}^j(t_{k+\Delta t}) | P_{zve}^j \end{array} \right\}.$$

7. Построение на основе синтезированной модели допустимых множеств состояний и функций глобальных моделей на различных уровнях  $\{DA^i, DB^i, DC^i, FB^i, FC^i\}$

8. Масштабирование результатов на интервалы. Рассматриваются интервалы различной длительности:  $[T_a; T_b]; [T_A; T_B] \subset [T_a; T_b] \subset [T_a; T_b]$ .

Для синтеза моделей объектов с применением предложенных методов необходимо наличие данных об объекте. Для получения необходимых данных выполняется синтез процессов мониторинга, на основе которых строятся программы мониторинга.

### 3.2. Метод многоуровневого синтеза автоматных моделей процессов мониторинга

*Постановка задачи многоуровневого дедуктивного синтеза автоматных моделей процессов мониторинга.* Задача синтеза многоуровневых процессов предусматривает, что заданы исходная модель  $\{d_s(t_k)\} = \{d_s^0(t_k)\}, \{d_s^1(t_k)\}, \dots, \{d_s^i(t_k)\}, \dots, \{d_s^L(t_k)\}$  и целевая модель  $\{d_w(t_{k+\Delta t})\} = \{d_w^0(t_{k+\Delta t})\}, \{d_w^1(t_{k+\Delta t})\}, \dots, \{d_w^i(t_{k+\Delta t})\}, \dots, \{d_w^L(t_{k+\Delta t})\}$ . Определены функции их связывающие

$$F_{zv} : \left\{ \begin{array}{l} F_{zve}^0(d_{zve}^0) | P_{zve}^0 \xrightarrow{P_{zve}^{0,j}(\bullet)} d_{zva}^j | P_{zve}^j \\ \dots \\ F_{zve}^L(d_{zve}^L) | P_{zve}^L \xrightarrow{P_{zve}^{L,j}(\bullet)} d_{zva}^j | P_{zve}^j \end{array} \right\}, \quad (3.7)$$

где  $F_{zve}^i(\bullet) \in FB^i, FC^i$ ;  $d_{zve}^i, d_{zva}^i \in DD^i$ ;  $e = \overline{1, E_z}$ ;  $z = \overline{1, Z}$ ;  $v = \overline{1, V}$ ;  $i, j = \overline{1, L}$ . Связи определены для условий  $P_{zve}^{i,j}(\bullet)$ .

Требуется синтезировать процесс  $\{d_s\} \rightarrow \{d_w\}$  для условий  $P_{zv}(\cdot)$ :

$$PR = \left\{ F_{zv} (d_{zv_e}; e = \overline{1, E_z}) \rightarrow d_{zv_a}; \{d_s\}; \{d_w\}; \{P_{zv}(\cdot)\}; z = \overline{1, Z}; v = \overline{1, V_z} \right\}. \quad (3.8)$$

*Метод многоуровневого синтеза автоматных моделей процессов мониторинга* Синтез процессов основан на доказательстве существования требуемых процессов. Метод описан в работе автора [164], [165]. Для доказательства применяется прямой вывод. Предлагаемый метод многоуровневого синтеза моделей процессов мониторинга включает следующие шаги:

1. Решение задачи синтеза на верхнем уровне  $i = L - 1$ .
2. Обработка логических условий для уровня  $P^i(\cdot)$ .
3. Доказательство существования перехода  $\{d_s^i\} \rightarrow \{d_w^i\}$ .
4. Добавление доказанных результатов к данным  $\{d_s^i\}$ , не доказанные результаты добавляются к целям  $\{d_w^i\}$ .

5. Для не доказанных целей обращение к условиям задачи на более низких уровнях  $i$ :  $\{d_s^{L-2}\} \xrightarrow{P^{L-2}(\cdot)} \{d_w^{L-2}\}, \dots, \{d_s^i\} \xrightarrow{P^i(\cdot)} \{d_w^i\}, \dots, \{d_s^0\} \xrightarrow{P^0(\cdot)} \{d_w^0\}; i = \overline{L-2, 0}$ . При каждом переходе на более низкий уровень осуществляется возврат на Шаг 2.

6. Проверка достижимости целей  $\{d_w\}$  по результатам синтеза.

7. Построение результирующих процессов по уровням:

$$\left\{ \begin{array}{l} PR = \left\{ F^{L-1} (d_e^{L-1}; e = \overline{1, E_z}) \rightarrow d_a^{L-1}; \{d_s^{L-1}\}; \{d_w^{L-1}\}; \{P^{L-1}(\cdot)\}; \right\}; \\ \dots \\ PR = \left\{ F^0 (d_e^0; e = \overline{1, E_z}) \rightarrow d_a^0; \{d_s^0\}; \{d_w^0\}; \{P^0(\cdot)\}; j = \overline{1, L-1} \right\} \end{array} \right\}.$$

8. Сведение процессов в общий процесс:  $PR = \left\{ F (d_{zv_e}; e = \overline{1, E_z}) \rightarrow d_a; \{d_s\}; \{d_w\} \right\}$ .

При отсутствии возможности доказательства рассматривается возможность изменения требований, предъявляемых к моделям объектов.

### 3.3. Методы многоуровневого синтеза автоматных моделей программ мониторинга

*Постановка задачи многоуровневого синтеза автоматных моделей программ мониторинга.* Задача синтеза многоуровневых программ мониторинга формулируется следующим образом. Задан процесс  $PR: PR = \left\{ F_{zv}^i (d_{zv_e}^i; e = \overline{1, E_z}); \{d_s^i\}; \{d_w^i\}; z = \overline{1, Z}; v = \overline{1, V_z} \right\}$ .

Требуется синтезировать соответствующую процессу PR программу PRG:

$$PRG = \left\{ U_{zv} (d_{zv_e}; e = \overline{1, E_z}) \rightarrow d_{zv_a}; \{d_w\}; \{d_s\}; z = \overline{1, Z}; v = \overline{1, V_z} \right\}, \quad (3.9)$$

где  $U_{zv}$  - структурный элемент программы PRG, соответствующий функции  $F_{zv}$ .

Синтез программ на основе доказанного процесса построен на реализации обратного вывода. Вывод осуществляется по схеме зеркальной прямому выводу. При обратном выводе результирующая структура  $\{d_w\}$ , полученная при синтезе процессов, рассматривается как исходная, а исходная структура  $\{d_s\}$  - как результирующая.

*Метод многоуровневого синтеза автоматных моделей программ мониторинга.* Метод синтеза программ мониторинга предусматривает выполнение следующих основных шагов.

1. Синтез структуры опорной программы на основе результатов доказательства:  
 $\{d_s^i\} \leftarrow \{d_w^i\}; i = \overline{L-1, 0}$ ;

2. Синтез опорной программы в соответствии с синтезированной структурой:  
 $\{U_{zv}^i(d_{zve}^i; e = \overline{1, E_z}) \rightarrow d_{zva}^i; z = \overline{1, Z}; v = \overline{1, V_z}; i = \overline{L-1, 0}\}$ , где  $U_{zv}^i$  - элементы программной структуры.

3. Формирование логических условий для элементов опорной программы:  $P_{zv0}^i(U_{zv0}^i(\bullet))$ ,  
 $P_{zv1}^i(d_{zv1}^i), \dots, P_{zva}^i(d_{zva}^i)$ .

4. Построение многоуровневой программы, учитывающей логические условия:  
 $PRG^i = \{U_{zv}^i(d_{zve}^i; e = \overline{1, E_z}) \rightarrow d_{zva}^i; \{P_{zv}^i(\bullet)\}; \{d_w\}; \{d_s\}\}$ .

5. Трансляция многоуровневой программы в программу нижнего уровня:  
 $PRG^i \rightarrow PRG^{L-1}$ .

6. Назначение результирующей программы:  $PRG = PRG^{L-1}$ . Результирующая программа представляет собой последовательность программных конструкций.

Рассмотренные алгоритмы синтеза обеспечивают построение формальных моделей. Для синтеза конкретных моделей на основе полученных об объектах данных предложены методы и модели трансформации данных.

### 3.4. Методы и модели многоуровневой трансформации данных мониторинга

*Постановка задачи многоуровневой трансформации данных мониторинга.* Данные об объектах, получаемые в ходе мониторинга, представляются в виде:  $\{d_s\} = \{d_s^0\} = \{\xi(t_k), \zeta(t_k)\}$ ;  $t_{k-\Delta t} \leq t_k$ , где  $\xi$  - контент,  $\zeta$  - контекст обработки,  $t_k$  - заданный момент времени. Для синтеза моделей объектов необходимо построить представления данных на различных уровнях:  $\{d_w\} = R\{d_w^l, t_k\}; l \leq L; t_{k-\Delta t} \leq t_k \leq t_{k+\Delta t}; [t_{k-\Delta t}, t_{k+\Delta t}] \in T_{набл}$ , где  $L$  - уровни модели;  $T_{набл}$  - период наблюдения. Построение представлений и переходов между ними предполагает выполнение множественных трансформаций данных.

Задача трансформации данных мониторинга формулируется следующим образом. Требуется определить последовательности трансформаций контента в контексте, позволяющих сформировать представления данных, необходимые для синтеза моделей объектов. Полученные последовательности трансформаций определяют процессы обработки данных вида:

$$P = \langle p_{rk}^j \mid C_{rk}^j \rangle; j = \overline{L, l}; r = \overline{1, R}; k = \overline{j, K}, \quad (3.10)$$

где  $p_{rk}^j$  -  $r$ -ый элемент процесса на шаг  $k$  решаемой задачи.

При таком построении процессы (3.10) являются контентно-адаптивными процессами. Построение процессов осуществляется с учетом имеющихся результатов обработки.

### 3.4.1. Метод многоуровневой трансформации данных

Предложен метод многоуровневой трансформации данных [176], [177]. Он позволяет строить общие последовательности трансформаций и представлять их в виде параметрических семейств процессов обработки.

Метод предусматривает:

1. Определение пространства построения процессов обработки, описание процессов обработки в общем виде. При этом определяются стратегии для достижения целей обработки, фазовое пространство и пространство управления обработкой, пространство функционалов, применимых для достижения цели обработки.

2. Построение семейств трансформаций данных для выбранной стратегии. При этом строятся множества допустимых правил управления трансформациями для выбранной стратегии, формируется общая структура системы трансформаций в виде последовательности управляемых переходов между различными представлениями данных.

3. Построение системы трансформаций данных о наблюдаемых объектах. Для этого определяются трансформации в соответствии с типами представлений данных. На их основе строятся многоуровневые горизонтально-вертикальные структуры трансформаций.

4. Построение параметрических семейств процессов обработки данных о наблюдаемых объектах. На основе построенных трансформаций с горизонтально - вертикальной структурой определяются семейства параметрических процессов обработки данных.

Для описания данных, их трансформаций и формируемых представлений предложена модель трансформаций.

### 3.4.2. Модель многоуровневой трансформации данных

Предлагаемая модель многоуровневой трансформации данных в развернутом виде представлена в работах автора [176], [177]. Она развивает идеи адаптивной обработки, используя модели и методы приведенные в [178], [179].

Исходное пространство трансформаций данных определяется допустимым контентом информационных потоков  $K = \{\xi\}$ , допустимым контекстом обработки  $\Pi = \{\zeta\}$ , множеством функционалов, применимых для обработки контента в контексте  $\Phi = \{\varphi\}$ , возможными стратегиями обработки  $\Sigma = \{\sigma\}$ . Под стратегией понимается совокупность правил выбора действий. Процессы, реализующие стратегию обработки  $\sigma$ , за конечное время  $t_k$  представляются в виде:

$$P = [\tilde{U}, T_{\zeta_t}, L], t < t_k; \tilde{U} = \cup U_F; U_F = (X, F, Y); F: X^h \times Y^{h-1} \rightarrow Y, \quad (3.11)$$

где  $U_F$  - элементарный процесс;  $X$  - фазовое пространство обработки в контексте;  $Y$  - пространство управления процессами обработки;  $F$  - правила управления процессами;  $h$  - глубина памяти;  $T_{\zeta_t}$  - параметрическое семейство трансформаций вида  $T: D \rightarrow D$ ;  $D$  - множество допустимых правил управления для заданной стратегии;  $L$  - параметры, определяющие уровни обработки;  $\zeta_t = \Psi(\xi, \zeta)$  - статистика процесса в контексте.

Семейства трансформаций, обеспечивающих достижение конечной цели на классе  $K \times \Phi \times \Pi$  за  $m$  итераций за счет управляемых переходов на множестве  $D$  определяются как

$$F_k = T_{\zeta_k} \otimes F_{k-1}, k = \overline{1, m}, \quad (3.12)$$

где  $F_k$  - правила управления на  $k$ -ом шаге;  $T_{\zeta_k}$  - состав трансформаций на  $k$ -ом шаге,  $\otimes$  - операция трансформации. Такой процесс можно рассматривать как процесс управляемого блуждания на множестве  $D$ .

Системы трансформаций определяются для типов представлений данных:

$$T = \{\mathbf{E}, T_{\mathbf{E}}, id_{\mathbf{E}}, \circ\};$$

$$T_{\mathbf{E}} = \{\varphi | \varphi(\sigma): E^n \rightarrow E, E \in \mathbf{E}; n \in \mathbb{N}\}, E \in \mathbf{E}; \quad (3.13)$$

$$Cart(\mathbf{E}) = \{E^n | E \in \mathbf{E}; n \in \mathbb{N}\}; E^n = E \times E \times \dots \times E;$$

где  $\mathbf{E}$  - множество типов представлений  $E$ ,  $\varphi$  - функционалы, выбираемые в соответствии со стратегией  $\sigma$ ;  $\circ$  - композиция функций,  $id_{\mathbf{E}}$  - единичный элемент,  $id_{\mathbf{E}} = \{1_E | 1_E(x) = x, \forall x \in E, \forall E \in \mathbf{E}\}$ . Формируемые представления данных образуют многоуровневую структуру, описывающую объект в виде результатов обработки исходных данных. Примерами представлений данных являются исходные временные ряды, расчетные характеристики отдельных составляющих объекта, интегральные характеристики, описывающие объект в целом.

Системы трансформаций представляют собой структуры вида:

$$T_{\mathbf{E}} = \{T_{\mathbf{E}}^G \cup T_{\mathbf{E}}^{GV} \cup T_{\mathbf{E}}^V\}, \quad (3.14)$$

где  $T_E^G$  - вертикальные,  $T_E^V$  - горизонтальные,  $T_E^{GV}$  - вертикально-горизонтальные трансформации.

Параметрические семейства вертикальных  $P_E^V$  и горизонтальных  $P_E^G$  процессов обработки обеспечивают построение многоуровневых моделей объектов на основе представлений данных, определяемых структурами  $T_E^G$  и  $T_E^V$ :

$$P_E^{G,V} = \{E \rightarrow A_E; A = \psi_\tau(\xi, \zeta); A \in A_E; \tau \in Z\}; \quad (3.15)$$

$$A: A^1 \triangleleft A^2 \triangleleft A^3 \triangleleft \dots \triangleleft A^K; \psi: \varphi \times A^{i-1} \rightarrow A^i,$$

где  $A_E$  - множество моделей объекта  $E$ ;  $A^i$  -  $i$ -ый уровень модели  $A$ ;  $\psi(\xi, \zeta)$  - функционалы обработки на признаковом пространстве  $S_f(A)$ ;  $\tau = (\tau_1, \tau_2, \dots, \tau_n)$  - параметры процессов обработки;  $\xi$  - контент;  $\zeta$  - контекст. Параметрические семейства горизонтально-вертикальных процессов определяются структурой  $T_E^{GV}$ . Они обеспечивают построение систем связанных между собой моделей:

$$P_E^{GV} = \{r_\tau(\xi, \zeta) \mid r: A \times A \times \dots \rightarrow A^m; m \in \mathbb{N}\}; \quad (3.16)$$

$$A^m = \{A_i^k \rightleftharpoons A_j^l; i, j = \overline{1, m}; k, l = \overline{1, K}\},$$

где  $A^m$  - система из  $m$  связанных моделей;  $r(\xi, \zeta)$  - функционалы обработки на признаковом пространстве  $S_f(A^m)$ .

Предложенная модель позволяет описывать различные процессы обработки данных, в том числе, процессы слияния данных.

Для решения задач слияния данных предложена логическая модель, являющаяся обобщением существующих моделей [180]–[183]. Для модели разработаны информационные критерии, позволяющие оценивать формируемые результаты [163], [184]–[186].

Описания построенных моделей, ориентированных на слияние данных, приведены в статьях [187]–[190].

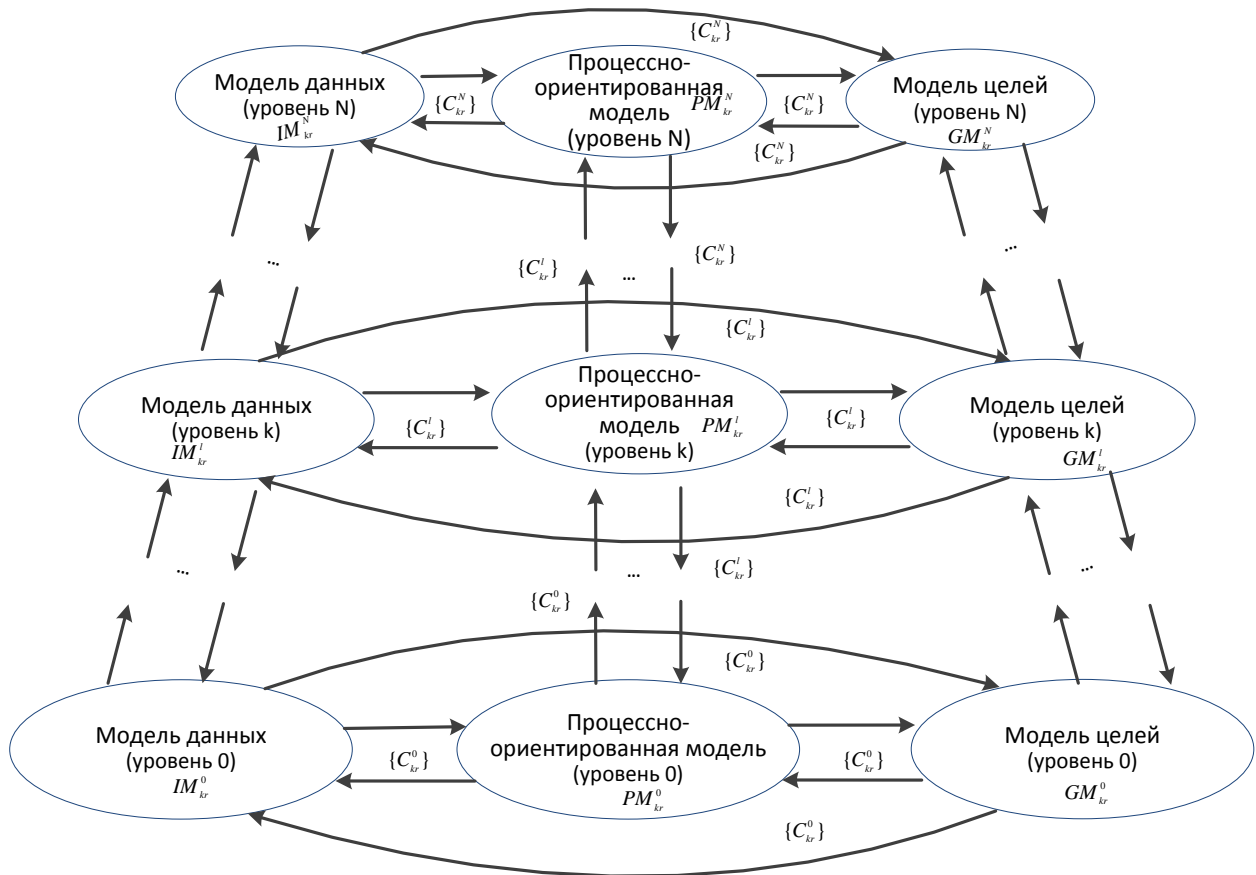
### 3.5. Методы и модели обеспечения трансформации данных мониторинга

Модели и методы обеспечения многоуровневой трансформации данных позволяют строить конечные процессы обработки данных в соответствии с предложенным методом трансформации данных. Для этого выполняется построение обобщенной модели обеспечения, ее детализация в виде системы связанных моделей, образующих комплексную модель. На основе комплексной модели формируются модели для программных систем. Предусмотрена модель управления, определяющая правила построения и перестройки моделей.



### 3.5.1. Обобщенная модель обеспечения трансформации данных мониторинга

Для представления формальных структур процессов обработки предложена модель, приведенная на рисунке 3.1. Модель имеет многоуровневую структуру. Уровни определяются в соответствии с представлениями данных. Структура уровней согласуется со структурой автоматных моделей. Для каждого уровня определяются модели данных, модели целей и процессно-ориентированные модели. Модели данных описывают исходные представления данных, модели целей - требуемые представления. Процессно-ориентированные модели определяют преобразования, позволяющие перейти от моделей данных к моделям целей с учетом условий существования переходов  $\{C_{kr}^l\}$ .



$\{(\bullet)_{kr}^l\}$  - элемент k-го уровня при решении задачи  $i$  на шаге  $j$ ;

$\{C_{kr}^l\}$  - условия существования переходов

Рисунок 3.1 – Обобщенная многоуровневая модель обеспечения трансформации данных

### 3.5.2. Комплексные модели обеспечения трансформации данных мониторинга

Обобщенная модель разворачивается в виде четырех связанных моделей: функциональной, информационной, процессной, сервисной (рисунок 3.2), образующих комплексную модель обеспечения трансформации данных. При их построении используется управляющая модель.

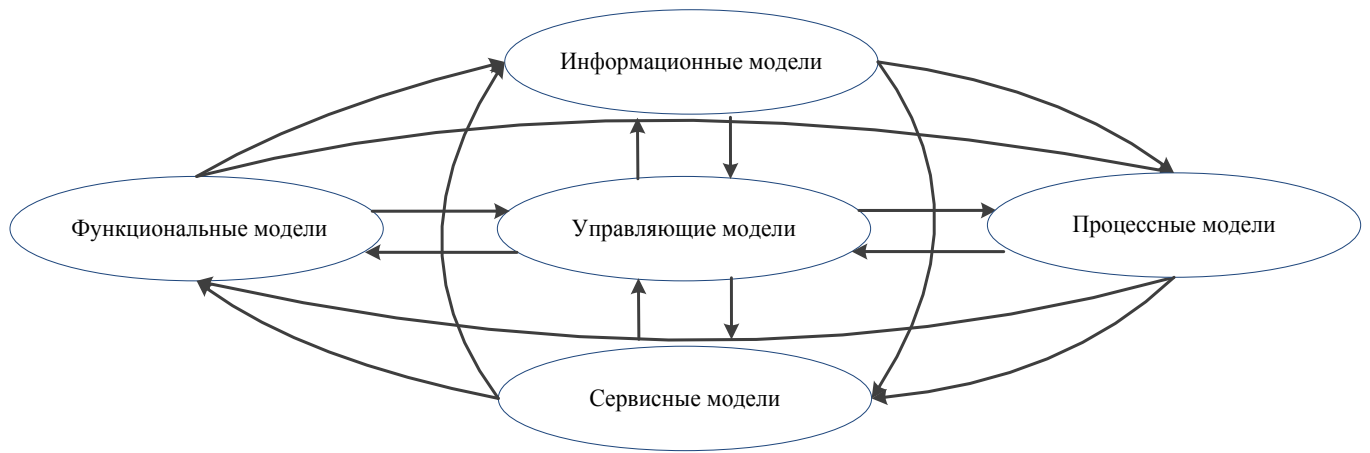


Рисунок 3.2 – Комплексная модель обеспечения трансформации данных

В комплексных моделях исходные данные и результаты их трансформаций на различных этапах обработки представляются в виде информационных моделей. Функциональные модели описывают частные задачи, решаемые на отдельных этапах обработки, состав необходимых для этого входных данных и ожидаемые результаты обработки. Процессные модели определяют состав преобразований, которые необходимо выполнить для решения поставленных задач обработки. Сервисные модели содержат данные о составе и возможностях программных систем и компонентов, которые позволяют реализовывать построенные процессы обработки. Управляющие модели определяют правила построения процессов обработки, построенные на основе системы классификаторов, включающей классификаторы исходных данных, результатов их обработки, средств обработки и другие.

В процессе обработки данных осуществляется параллельное построение всех моделей. При построении комплексных моделей выполняются следующие шаги:

- по данным, предоставляемым информационными моделями, выполняется детализация задач в функциональных моделях;
- в соответствии с задачами, определенными в функциональных моделях, и данными информационных моделей строятся процессные модели;
- на основе процессных, функциональных и информационных моделей определяются сервисные модели;
- получаемые результаты обработки позволяют определять новые уровни информационных моделей и детализировать существующие.

Ниже рассматриваются отдельные модели и методы их построения. Полное описание комплексных моделей приведено в монографиях автора [191], [192].

*Информационные модели обеспечения трансформации данных.* Для обработки необходимо определить структуры, в которых должны размещаться данные о наблюдаемых объектах. На основе данных об объектах и результатах их обработки могут строиться математические, логические и статистические модели. Модели могут характеризовать

отдельные параметры объектов, объекты в целом, ситуации, предметную область. Подробное описание информационных моделей, их основных структурных элементов и взаимосвязей между ними приведено в [193].

Значительная часть данных, передаваемых с объектов, представляет собой результаты измерений параметров объектов. Для их обработки применяются статистические методы и методы интеллектуального анализа данных. Для описания результатов предложена информационная модель представления результатов обработки. Модель впервые предложена в [194]. Она включает три взаимосвязанные модели: модель представления временных рядов результатов измерений параметров объектов, модель представления количественных и качественных данных о параметрах объектов, а также модель представления результатов совместной обработки данных различных типов. Ниже более подробно рассмотрена модель представления временных рядов результатов измерений параметров объектов.

Модель представления временных рядов результатов измерений включает четыре уровня.

Нулевой уровень модели. Модель представления временных рядов результатов измерений параметров. На нулевом уровне описывается поток данных, содержащий исходные результаты измерений, и измерения, получаемые после предварительной обработки данных. Модели, относящиеся к нулевому уровню и вопросы их построения рассмотрены в [195]–[197].

Исходные потоки данных представляют собой структурированные бинарные потоки. Для их описания предложены две основные модели – базовая (структурная) и графовая. Базовая модель описания структурированного бинарного потока представляет собой универсальное описание структуры потока. Графовая модель позволяет представить структуру потока в виде совокупности связанных элементов, для каждого из которых определен набор характеристик. При таком представлении обработка бинарных потоков может осуществляться с использованием имеющихся алгоритмов и инструментов работы с графами.

Структурная модель исходных потоков представляется как:  $G = \langle P \Omega F \rangle$ , где  $P = (P_1 \dots P_N)$  – множество измерений, передаваемых в потоках данных,  $\Omega$  – последовательность форматов, которые используются при описании потока,  $F$  – функция, позволяющая назначить позиции каждому из измерений. В случае, если при описании потока используется один формат, то структуру потока определяет пара  $\langle P F \rangle$ .

Измерения представляют собой ряды значений вида:  $P_i = p_i(t_1), \dots, p_i(t_n)$ ,  $t_1 < t_2 < \dots < t_n$ , где  $p_i(t_j)$  – значение  $i$ -го ряда в момент времени  $t_j$ . Последовательность форматов – это кортеж

$\Omega = (S_1 \dots S_l)$ , где  $S_l$  - очередной формат потока, а  $l$  - количество различных используемых форматов.

Описание формата кадра включает длину информационного слова  $L_{ИС}$ , длину кадра  $L_K$ , а также множества субкоммутиаций  $Sub$  и множество суперкоммутиаций  $Sup$ ,  $S = (L_{ИС} L_K Sub Sup)$ . Субкоммутиация используется для передачи нескольких параметров в одном слове, а суперкоммутиация – для передачи нескольких измерений одного параметра в одном кадре. Множество суперкоммутиаций  $Sup$  - это множество пар  $Sup = \{ (i f)_{0..k} \}$ , где  $i$  – индекс слова в кадре;  $f$  – кратность суперкоммутиации;  $k$  – число суперкоммутиаций в кадре. Множество субкоммутиаций  $Sub$  - это множество пар  $Sub = \{ (j d)_{0..m} \}$ ,  $j$  – индекс слова в кадре;  $d$  – глубина субкоммутиации;  $m$  – число субкоммутиаций в кадре.

Функция типа слова – это функция  $type(i) \in \{type_{Par}, type_{Sup}, type_{Sub}\}$ , которая определяет тип  $i$ -го слова в кадре:

$$\begin{cases} type(i) = type_{Sub}, \text{ если } \exists (i d) \in Sub \\ type(i) = type_{Sup}, \text{ если } \exists (i f) \in Sup \\ type(i) = type_{Par}, \text{ если } \forall f : (i f) \notin Sup, (i f) \notin Sub \end{cases}$$

Здесь метка Par означает параметр, передаваемый без суб- и супер- коммутиаций, Sup – наличие в данном слове суперкоммутиации, Sub – наличие субкоммутиации.

Функция типа слова  $type(i)$  и пара множеств  $Sup$  и  $Sub$  являются взаимозаменяемыми, поэтому формат кадра можно представить следующим образом:  $S = \langle L_{ИС} L_K type \rangle$ , где  $type$  – определенная выше функция типа слова в кадре. Тогда базовая модель может быть представлена следующим образом:  $G = \langle PSF \rangle$ , где  $P = \langle P_1 \dots P_N \rangle$  – множество передаваемых параметров;  $S$  – формат кадра;  $F$  – функция назначения позиции каждому параметру.

Графовая модель структурированного потока представляет собой нагруженный граф отношений (НГО). НГО определяется как:  $G = \langle V E A_V A_E \alpha_V \alpha_E \rangle$ , где  $V$  – множество вершин НГО;  $E$  - множество ребер НГО;  $A_V$  – множества атрибутов вершин;  $A_E$  - множество атрибутов ребер;  $\alpha_V$  – функции назначения атрибутов вершинам;  $\alpha_E$  - функции назначения атрибутов ребрам. Атрибуты вершин и ребер имеют вид:  $t(p_1, \dots, p_{k_t})$ ,  $t \in T$ ;  $t$  - тип атрибута,  $T$  - элемент конечного алфавита  $T$ ;  $(p_1, \dots, p_{k_t})$  - параметры атрибута, заданные на конечных множествах  $P_1, \dots, P_{k_t}$ . Число атрибутов  $k_t$  и их последовательность зависят от типа атрибута.

Существуют особые типы атрибутов, которые не имеют параметров. Они используются для задания различных типов узлов и ребер.

Граф, описывающий бинарный поток, имеет древовидную структуру, которая отражает структуру соединений датчиков и коммутаторов, установленных на объектах. Для описания вершин, соответствующих датчикам, и вершин, соответствующих коммутаторам, используются атрибуты различных типов:  $T_V = \{d, c\}$ , где  $d$  – тип атрибута «датчик»;  $c$  – тип атрибута «коммутатор». Атрибуты ребер относятся к особому типу атрибутов, которые не имеют параметров:  $T_E = \{t^0\}$ .

В формируемой структуре субкоммутаторы представляются в виде узловых вершин, главному коммутатору сопоставляется корневой элемент дерева, листья дерева содержат описания параметров, измеряемых конечными датчиками.

Коммутаторы характеризуются последовательностью опроса подключенных к нему датчиков и субкоммутаторов. С учетом этого каждый коммутатор описывается кортежем имен вершин-потомков  $nv_1 \dots nv_m$ ;  $P_c = \langle nv_1 \dots nv_m \rangle$ , где  $nv_1 \dots nv_m$  – имена вершин  $v_1 \dots v_m \in V$ .

Для описания листьев дерева, соответствующих измеряемым параметрам, строится гистограмма частоты встречаемости значений информационных слов для рассматриваемого параметра:  $P_D = \langle p_0 \dots p_l \rangle$ , где  $l$  – число возможных информационных слов,  $l = 2^{L_{ис}}$ ;  $p_0 \dots p_l$  – вероятности появления значений  $0 \dots 1$ .

Графовая модель структурированного потока имеет вид:  $GTS = \langle L_{ис} \ M \ V \ A_E \ \alpha_E \rangle$ , где  $A_E = \begin{bmatrix} c \langle nv_1 \dots nv_m \rangle \\ d \langle p_0 \dots p_l \rangle \end{bmatrix}$ .

Переход к обобщенной графовой модели, позволяющей описывать бинарные потоки со сменой формата кадра, осуществляется за счет рассмотрения нового типа ребер, фиксирующего связи замещения исходного элемента структуры потока на новый. В обобщенной модели множество атрибутов ребер описывается в виде  $T_E^+ = \{e^0, r^0\}$ , где  $e^0$  – атрибут ребер, отражающий непосредственную связь элементов потока исходного и нового форматов;  $r^0$  – атрибут ребер, устанавливаемый в случае замещения элемента потока исходного формата другим элементом в новом формате. Возможен случай, когда после смены структуры кадра параметр, оставаясь на  $i$ -ой позиции, характеризует состояние другого элемента объекта. Такие параметры в графовой модели представляются в виде 2х узлов, которые характеризуют параметры до и после смены формата кадра.

Исходные результаты измерений, полученные из потоков, представляют собой множество элементов  $C = \{c_n\}_{n=1}^N$ , где  $c_n = \{t_n, x_n\}_{n=1}^N$ ,  $x_n$  - дискретное измерение, выполненное в момент времени  $t_n$ ,  $N$  - общее количество измерений. Рассмотренное множество пар является упорядоченным по временным отсчётам множеством  $\min(I) \leq t_1 \leq t_2 \leq \dots \leq t_N \leq \max(I)$ , где  $I$  - интервал измерения. Предполагается, что интервалы между измерениями являются равными:  $\Delta t_i = t_i - t_{i-1}$ ;  $\Delta t_2 = \Delta t_3 = \dots = \Delta t_n$ . Для пространственно соотнесенных временных рядов измерений  $c_n = \{t_n, p_n, x_n\}_{n=1}^N$ , задаются пространственные координаты  $p_n$ .

Формат представления данных после предварительной обработки аналогичен формату представления исходных измерений, однако дополняется информацией о выявленных трендах и смещениях. Кроме того, в ходе предварительной обработки выявляются измерения, которые не содержат информации об исходном процессе и подлежат исключению, восстанавливаются пропущенные значения, удаляются одиночные и групповые выбросы, устраняется шум и т.д.

Первый уровень модели. Модели структурного представления результатов измерений. Модели и отдельные методы их построения рассмотрены в [198]–[200]. Предобработанная исходная последовательность измерений  $\{k_n\}_{n=1}^N$  представляется в виде последовательности блоков (сегментов). При этом значения характеристик временного ряда в пределах одного блока не изменяются, но они имеют резкие скачки в точках переходов, которые соответствуют краям блока. Блоки формируются из ячеек данных, каждый блок представляет собой набор из соседних ячеек:  $B(n, m) = \{K_n, K_{n+1}, \dots, K_m\}$ . Такой блок содержит  $m-n+1$  ячейку, начиная с  $n$ -ой. Тогда исходная последовательность измерений или некоторый ее интервал  $I$  - это набор непересекающихся блоков. Разбиение может быть описано количеством блоков  $N_{blocks}$ , и набором граничных ячеек  $n_k: P(I) \equiv \{N_{blocks}, n_k, k = 1, 2, 3, \dots, N_{blocks}\}$ . Общее количество вариантов разбиения некоторой последовательности значений на блоки:  $N_{partitions} = 2^{N-1}$ .

Множество сегментов, выделяемых во временных рядах результатов измерений, разделяется на три подмножества:  $S = \{S_T, S_P, S_S\}$ , где  $S_T$  - типовые сегменты;  $S_P$  - шаблонные сегменты;  $S_S$  - специальные сегменты. К типовым сегментам относятся: константные, линейно возрастающие, линейно убывающие, выпукло возрастающие, выпукло убывающие, вогнуто возрастающие, вогнуто убывающие сегменты. Они описывают сегменты константными значениями, коэффициентами наклона или коэффициентами полинома второй степени. Среди типовых сегментов выделяются простые и составные. Шаблонные сегменты определяются априорно и зависят от свойств источника данных. Типовые шаблоны, как правило,

формируются для фрагментов временных рядов, характерных для одного из состояний источника данных. Под специальными сегментами понимаются сегменты, не входящие в состав типовых и шаблонных сегментов. Они выявляются в ходе сегментации временных рядов.

Для каждого из сегментов определяется и рассчитывается состав характеристик, которыми возможно описать поведение временного ряда на этом интервале. Полученный перечень характеристик определяет модель сегмента:  $S = \langle M_1(\alpha_1, \beta_1, \dots, \omega_1), \dots, M_k(\alpha_k, \beta_k, \dots, \omega_k) \rangle$ , где  $M_i(\alpha_i, \beta_i, \dots, \omega_i)$  — описание поведения временного ряда на  $i$ -м интервале с помощью характеристик  $\alpha_i, \beta_i, \dots, \omega_i$ . Длина участка, на котором сохраняется модель поведения  $M_i$  с набором характеристик  $\alpha_i, \beta_i, \dots, \omega_i$ , обозначается как  $|M_i|$ .

Сегменты временных рядов группируются по классам состояний. Пусть  $S$  – исходные измерения, содержащие  $n$  отсчетов. Пусть  $C$  – множество всех возможных различных состояний источника измерений. Пусть некоторое состояние  $c \in C$  наблюдается в течение периода времени  $[b, f]$ , где  $b$  и  $f$  обозначают соответственно начальный момент времени, когда начинает наблюдаться состояние  $c$ , и заключительный момент времени, когда состояние  $c$  больше не наблюдается. Тогда  $c$  рассматривается как описание интервала  $[b, f]$ .

Второй уровень модели. Модели содержательного представления результатов измерений [201]. При формировании алфавитного представления временных рядов результатов измерений каждому сегменту ставится в соответствие буква из некоторого конечного алфавита  $t \in T$ . При выборе буквы, как правило, учитываются значения рассчитанных характеристик, класс сегмента. В результате, временной ряд представляется в виде последовательности символов.

На основе символьного представления определяются типы временных рядов. Каждый тип временного ряда описывается эталонным поведением  $E$ . Эталонное поведение задается в виде одного или нескольких временных рядов и соответствующих им символьных представлений:  $E = E_1, \dots, E_m$ , где  $E_i$  — временной ряд  $e_1^i, \dots, e_n^i$ , задающий эталонное поведение, и его символьное представление,  $m$  — количество возможных вариантов поведения временных рядов, принадлежащих типу.

Шаблоны временных рядов измерений строятся на основе алфавитного представления временных рядов или на основе множества выделенных сегментов, для которых определены классы состояний. Шаблоны могут рассматриваться как обобщенное описание временных рядов. Шаблоны представляется в виде упорядоченных пар  $P = (A, R)$  или  $P = (C, R)$ , где  $C$  – множество классов состояний,  $A$  – алфавит,  $R$  – множество допустимых связей (закономерностей в следовании) между состояниями. Множество возможных связей

ограничивается выразительными возможностями темпоральной логики. Для  $n$  интервалов  $(b_i, f_i, s_i)_{1 \leq i \leq n}$  их взаимное положение описывается с помощью матрицы  $n \times n$ , каждый из элементов которой  $RM[i, j] \in R$  определяет отношение между интервалами за номерами  $i$  и  $j$ .

Образы временных рядов строятся для описания групп схожих временных рядов, например, временных рядов результатов измерений одного параметра объекта [202]. Обобщенный образ временного ряда представляется в виде множества:  $O = \langle O^+, O^- \rangle$ , где  $O^+ = \{O_s\}_k$  - образ временного ряда, включающий набор частных образов, описывающих типовое поведение временных рядов в рассматриваемой группе;  $O^- = \{Pr\}_l$  - образ, содержащий набор временных рядов, отражающих не типовое поведение временных рядов в группе;  $O_s$  - частный образ временного ряда, включающий набор шаблонов и эталонных временных рядов;  $PR$  - реализация временного ряда измерений;  $k$  - количество частных образов;  $l$  - количество временных рядов.

Третий уровень модели. На основе шаблонов измерений строится модель представления результатов измерений в виде графа переходов между состояниями объекта. Она представляется в виде упорядоченной пары  $P = (C, R)$ , где  $C$  - множество классов состояний объекта,  $R$  - множество допустимых связей (закономерностей в следовании) между ними.

Зависимости в поведении временных рядов результатов измерений представляются с использованием логических правил, в частности, ассоциативных правил вида  $X \Rightarrow Y$ ,  $X \cap Y = \emptyset$ , где  $X \subset C$ ,  $Y \subset C$  для случая, когда измерения представлены в виде последовательности наборов состояний; для случая, когда используется алфавитное представление параметров  $X \subset T$ ,  $Y \subset T$ . Правила  $X \Rightarrow Y$  характеризуются значением поддержки  $\text{supp}$ :  $\text{supp}(X \Rightarrow Y) = \text{supp}(X \cup Y)$  и достоверности  $\text{conf}$ :  $\text{conf}(X \Rightarrow Y) = \text{supp}(X \cup Y) / \text{supp}(X)$ .

Граф переходов между состояниями записывается с помощью набора ассоциативных правил, при этом для каждого правила указывается тип отношения  $R$ :  $X \Rightarrow \text{relation} Y$ , где  $X \subset C$ ,  $Y \subset C$  и  $X \cap Y = \emptyset$ .

*Функциональные модели обеспечения трансформации данных.* Функциональные модели определяют состав задач обработки данных. В состав функциональных моделей включены модели, относящиеся к разным уровням обработки данных: модель согласования форматов и структур данных; модель структурной и содержательной интеграции данных; модель интеллектуального анализа данных; модель интерпретации данных; модель исследования данных. Функциональным моделям обработки данных посвящены статьи автора [203], [204].

*Процессные модели обеспечения трансформации данных.* Процессные модели определяют преобразования, которые необходимо выполнить над данными для решения



поставленных задач обработки. Формируемые процессы являются контентно-адаптивными процессами. Модели и методы их построения рассмотрены в работах автора [205]–[209].

Общая схема построения процессов контентно-адаптивной обработки показана на рисунке 3.3. Она предусматривает построение моделей процессов обработки данных структура которых приведена на рисунке 3.4.

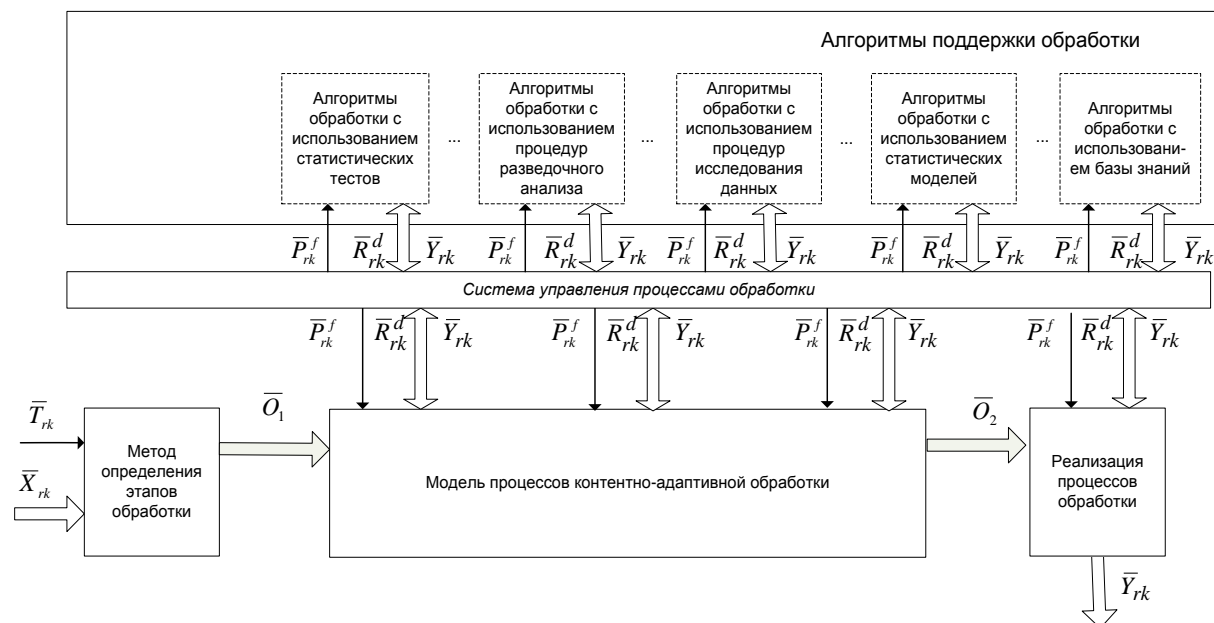


Рисунок 3.3 – Схема выполнения процессов контентно адаптивной обработки

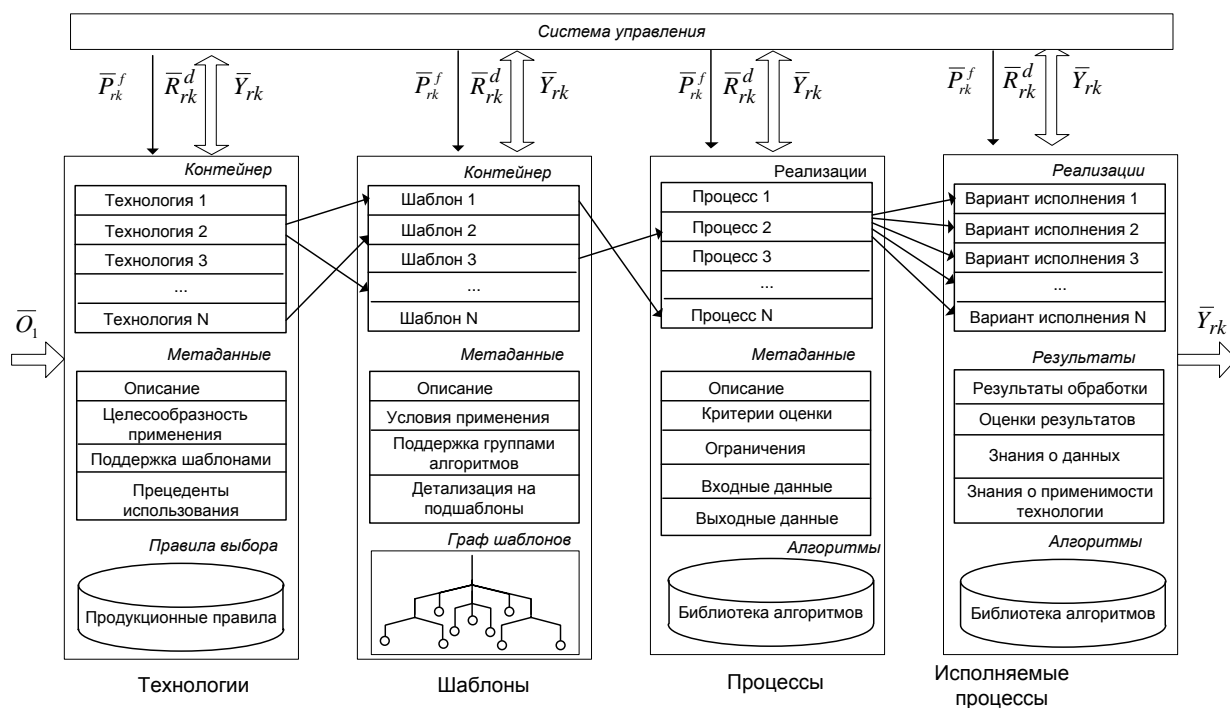


Рисунок 3.4 – Структура моделей процессов контентно-адаптивной обработки

При построении первого уровня моделей выполняется декомпозиция решаемой задачи на набор подзадач для которых определяются общие подходы к их решению. Для каждой

задачи функциональными моделями определяется перечень необходимых исходных данных, получаемых результатов, критериев их оценки. Каждая из подзадач, в свою очередь, также может быть декомпозирована на набор подзадач, в результате чего формируется иерархия. В соответствии с задачами и подзадачами определяется иерархия этапов и подэтапов обработки.

При формировании второго уровня моделей осуществляется выбор целесообразных технологий обработки. Технологии представляют собой обобщенные методы решения задач и описываются в виде множества  $W = \langle I, C, W^R, z_w \rangle$ , где  $I$  - этап обработки,  $C$  - множество технологий обработки,  $C = \{J\}_{i=1}^{N_C}$ ,  $J = \langle d_1, \dots, d_k, \dots, d_{N_J} \rangle$ ,  $J$  - технология обработки, обеспечивающая решение одной из задач обработки,  $d_k$  - действие по обработке, реализуемое в рамках технологии,  $W^R$  - множество правил выбора технологий обработки,  $N_C$  - общее количество технологий обработки,  $N_J$  - общее количество действий по обработке, предполагаемое  $J$ . Функция  $z_w$  обеспечивает выбор технологии обработки из множества имеющихся технологий с использованием правил  $W^R$ ,  $z_w : \langle C, W^R \rangle \rightarrow J$ .

На третьем уровне определяются возможные варианты реализации технологий. Такие варианты представляются в виде множества связанных шаблонов обработки. Под шаблонами обработки понимаются детальные описания реализации технологий, представляемых в виде графов. В их узлах могут размещаться шаблоны или группы алгоритмов обработки. Для формального описания шаблонов применяются ориентированные атрибутированные графы отношений  $G$ . Пусть  $G_V^A$  - это атрибуты вершин,  $G_E^A$  - атрибуты ребер. Каждый атрибут представляется в виде  $(\sigma, \varphi)$ , где  $\sigma$  - тип атрибута,  $\varphi = (\varphi_1, \dots, \varphi_{N_\sigma})$  - вектор характеристик, ассоциированный с  $\sigma$ ,  $N_\sigma$  - количество элементов вектора характеристик. Тогда атрибутированный граф отношений над множеством атрибутов  $G^A = G_V^A \cup G_E^A$  есть множество  $G = \langle J, G_V, G_E, z_{G_E}, z_{G_V} \rangle$ , где  $J$  - технология обработки, для которой строится шаблон,  $G_V$  - множество вершин графа,  $G_V = \langle b, r \rangle$ ,  $b$  - тип вершины,  $b \in G_V^A$ ,  $r$  - предусловия и постусловия выполнения обработки,  $G_E \subseteq G_V \times G_V$  - множество ребер,  $z_{G_V} : G_V \rightarrow G_V^A$  - функция назначения атрибутов вершинам,  $z_{G_E} : G_E \rightarrow G_E^A$  - функция назначения атрибутов ребрам. Выделяется два типа атрибутов вершин  $G_V^A = \{b_a, b_p\}$ , где  $b_a$  - вершина, которая содержит именованную группу алгоритмов обработки,  $b_p$  - вершина, содержащая именованный шаблон обработки. Атрибуты для вершин типа  $b_a$  задаются в виде набора алгоритмов, которые могут быть применены:  $A = \{a_1, \dots, a_{N_A}\}$ , где  $a_i$  -  $i$  алгоритм обработки,  $N_A$  - общее количество

алгоритмов обработки. Атрибуты для вершин типа  $b_p$  задаются в виде набора шаблонов, которые могут быть применены. Множество атрибутов ребер имеет вид  $G_E^A = \{g_1^R, \dots, g_{N_{G^R}}^R\}$ , где  $g_i^R$  -  $i$  правило перехода из текущей вершины в следующую вершину,  $N_{G^R}$  - общее количество правил перехода.

На третьем уровне моделей строятся процессы обработки. При этом из каждой группы алгоритмов выбирается один или несколько алгоритмов, исходя из свойств обрабатываемых данных и факторов, оказывающих влияние на обработку. Формируемые процессы описываются в виде множества  $K = \langle G, A^d, O, z_d, z_a, z_c \rangle$ , где  $I$  - шаблон, на основании которого сформирован процесс обработки,  $O = \langle a_1^d, \dots, a_k^d \rangle$  - последовательность описаний алгоритмов, последовательное или параллельное выполнение которых предполагается в рамках выполнения процесса. Описание алгоритмов включают базовые сведения об алгоритме, описание условий применения и критерии оценки  $a_i^d = \{a_i, r_i, p_i\}$ ,  $i=1 \dots N_A$ ,  $a_i$  -  $i$  алгоритм обработки (наименование, ссылка на реализацию),  $p_i$  - предусловия выполнения  $a_i$ ,  $s_i$  - постусловия выполнения  $a_i$ . Последовательность выполнения алгоритмов определяется с использованием функции  $z_d$  отображения шаблона обработки в процесс обработки. Функция  $z_a : A^d \rightarrow \{x_p, x_u\}$  определяет тип выполнения алгоритмов:  $x_p$  - последовательное выполнение,  $x_u$  - параллельное выполнение. Функция  $z_c$  определяет условия, при которых допустимо последовательное / параллельное выполнение алгоритмов.

На четвертом уровне моделей строятся исполняемые процессы обработки данных. Исполняемые процессы - это процессы, для которых определены все параметры их выполнения. Они представляются в виде множества:  $K^E = \langle K, C, z_o \rangle$ , где  $K$  - процесс,  $K^E$  - исполняемый процесс,  $E = \{e_i\}_{i=1}^{N_E}$ ,  $e_i = \{a_i, d_i\}$ ,  $e_i$  -  $i$ -ый элемент исполняемого процесса,  $a_i$  -  $i$ -ый алгоритм обработки,  $d_i$  - множество параметров  $i$ -го алгоритма обработки,  $N_E$  количество элементов исполняемого процесса. Функция  $\theta$  определяет отображение каждого элемента процесса в элемент исполняемого процесса  $\theta : K \rightarrow K^E$ .

*Сервисные модели обеспечения трансформации данных.* Сервисные модели позволяют описывать процессы преобразования данных в терминах сервисов. При этом процессы определяются как последовательности обращений к сервисам, каждый процесс при обращении к нему рассматривается как сервис. Более подробно сервисные модели описаны в работах автора [210], [211].

Управляющие модели обеспечения трансформации данных. Управление предусматривает построение и перестройку структур моделей в соответствии с предъявляемыми к ним требованиями и условиями мониторинга [212]. Структура управляющей модели представлена на рисунке 3.5.

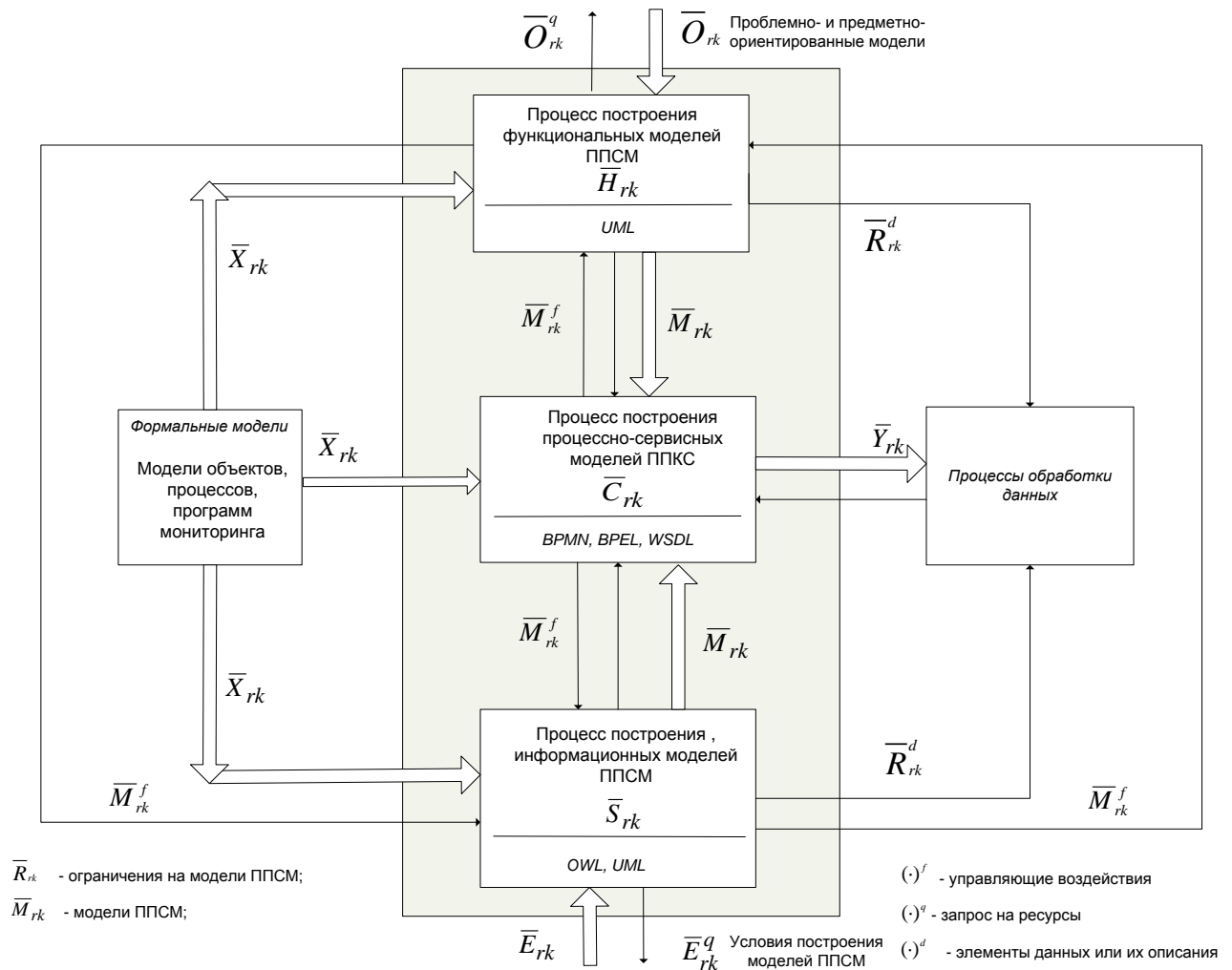


Рисунок 3.5 – Структура управляющей модели

Согласно представленной структуре требования к процессам обработки определяются синтезированными формальными моделями  $\bar{X}_{rk}$ , которые построены при решении  $k$ -ой задачи на момент времени  $t$ . Исходя из предъявляемых требований строятся функциональные ( $\bar{H}_{rk}$ ) и информационные ( $\bar{S}_{rk}$ ) модели ППСМ. При построении моделей учитываются имеющиеся предметно и проблемно ориентированные модели ( $\bar{O}_{rk}$ ) и условия построения моделей ( $\bar{E}_{rk}$ ). Возможно получение дополнительной информации из внешней среды за счет формирования запросов  $\bar{O}_{rk}^q$  и  $\bar{E}_{rk}^q$ . На основе функциональных и информационных моделей строятся процессные и сервисные модели ( $\bar{C}_{rk}$ ). В соответствии с ними формируются процессы

обработки данных  $\bar{Y}_{rk}$ . На процессы обработки могут накладываться ограничения ( $\bar{R}_{rk}^d$ ), определяемые со стороны функциональных и информационных моделей.

Таким образом, состояния управляющих моделей на момент времени  $t$  определяются состояниями функциональных, информационных, процессных и сервисных моделей  $\bar{B}_r = \{\bar{H}_r, \bar{S}_r, \bar{C}_r\}$ . При изменении входных параметров выполняется перестройка моделей, приводящая к перестройке процессов обработки данных. Вопросы учета условий  $\bar{E}_{rk}$  при перестройке рассмотрены в [213]–[215]. Примеры моделей  $\bar{O}_{rk}$  приведены в [216]–[218].

Для формирования воздействий по построению и перестройке моделей разработано множество связанных классификаторов в состав которых включены: классификатор временных рядов, классификатор сегментов временных рядов, классификатор методов, алгоритмов и процедур обработки и анализа данных, классификатор характеристик данных, классификатор вариантов формализованного описания данных, классификатор критериев оценки данных и результатов работы алгоритмов, классификатор формализованного представления данных и результатов их обработки, классификатор состояния объектов, классификатор ситуаций и др. Полный перечень предложенных классификаторов и их описания приведены в [177]. Отдельные классификаторы моделей и методов интеллектуальной обработки данных рассмотрены в статье [188], информационные критерии оценки результатов обработки приведены в [185]. В статьях также рассмотрены существующие классификаторы, определено их место в построенной системе классификаторов. Ниже рассмотрены отдельные классификаторы.

*Классификатор временных рядов.* Классификатор строится для временных рядов измерений параметров наблюдаемых объектов. Классификация временных рядов измерений основана на формировании групп временных рядов в соответствии с их поведением. Группы определяются с точки зрения методов и средств, которые могут быть применены для их обработки. Ниже рассматривается перечень классификационных признаков временных рядов.

1. В зависимости от характера изменения во времени параметры разделяются на функциональные, сигнальные и константные. Характерная особенность функциональных параметров состоит в том, что они являются непрерывными во времени функциями. Для сигнальных параметров характерно скачкообразное изменение во времени, например, связанное с переходом из одного дискретного состояния в другое. Константные параметры описываются одним значением.

2. В зависимости от скорости изменения во времени функциональные параметры разделяются на медленно меняющиеся и быстро меняющиеся. Первые характеризуются

спектром частот от 0 до 20-50 Гц, частота измерения вторых достигает 2.3 кГц и более. Медленно меняющиеся параметры наиболее многочисленны.

3. По характеру поведения временных рядов результатов измерений выделяются стационарные, нестационарные и кусочно-стационарные временные ряды. Временные ряды измерений независимо от их физической природы в большинстве случаев являются случайными нестационарными процессами. Однако, для них могут быть определены относительно протяженные участки стационарности.

4. Для временных рядов измерений медленно меняющихся функциональных параметров определяется наличие или отсутствие разрывов в первой и второй производных.

5. Для временных рядов измерений функциональных параметров определяется возможность их описания параметрическими моделями. Для нестационарных рядов подбор модели осуществляется для каждого из стационарных участков ряда. Состав моделей определяется в соответствии с существующими статистическими и интеллектуальными моделями описания временных рядов. Для оценки соответствия модели и временного ряда в типовом случае используется метод наименьших квадратов, возможно использование любого другого метода, например, метода максимального правдоподобия.

6. Наиболее распространенным является способ классификации временных рядов, основанный на вычислении совокупности различных характеристик исходного временного ряда. Состав рассчитываемых характеристик определяется соответствующим классификатором. Использование совокупности признаков для классификации временных рядов предполагает применение двух ступенчатой процедуры – построение классификатора средствами методов машинного обучения и его последующее использование.

*Классификатор сегментов временных рядов.* Сегментирование осуществляется для кусочно-стационарных и нестационарных временных рядов. При классификации сегментов временных рядов используются следующие способы определения типов сегментов:

-проводится анализ диапазона значений одной или совокупности рассчитанных для сегмента характеристик. Например, временные ряды значений константных параметров могут описываться медианой, разбросом значений, а также процентным соотношением значений, совпадающих со значением медианы и отличных от нее. При описании сигнальных параметров могут рассматриваться времена переходов и значения уровней для этих переходов;

- осуществляется поиск схожего сегмента в априорно сформированной базе типовых сегментов. В базу типовых сегментов могут включаться сегменты, соответствующие константному, линейно возрастающему / убывающему, выпукло / вогнуто возрастающему / убывающему поведению временных рядов. В процессе накопления исторических данных база сегментов расширяется за счет добавления в базу специализированных сегментов, отражающих

характерные особенности поведения временных рядов результатов измерений параметров различных объектов. Для сравнения сегментов вычисляется расстояние между ними.

При реализации второго варианта целесообразным является использование мультиразрешающего подхода к представлению временных рядов. В соответствии с ним верхний уровень представления отражает общие тенденции в поведении временного ряда. Затем определяются уровни, на которых учитываются только отдельные особенности его поведения. На нижнем уровне представления учитываются все выявленные локальные изменения. При описании сегментов формируется общий вектор, включающий характеристики временного ряда, полученные на различных уровнях.

*Классификатор характеристик временных рядов.* Характеристики ориентированы на описание стационарных, кусочно-стационарных и нестационарных временных рядов, а также их сегментов.

Классификатор характеристик временных рядов предполагает формирование групп характеристик по скорости их вычисления и области представления временных рядов результатов измерений (временная, частотная, частотно-временная), а также степени информативности характеристик для некоторой предметной области.

К первой группе характеристик относятся: частота измерений, значения статистических характеристик (медиана, мода, размах, ранг, среднее квадратическое отклонение, коэффициент вариации, моменты (математическое ожидание, дисперсия, асимметрия, эксцесс), поведение кривой, соответствующей временному ряду во временной области (вариабельность, ошибка кусочно-постоянной аппроксимации, ошибка кусочно-линейной аппроксимации, ошибка аппроксимации полиномом второй и более степеней, значения характерных точек, кривизна), поведение кривой в частотной области (вектор коэффициентов Фурье), поведение кривой в частотно-временной области (вектор вейвлет-коэффициентов), энтропия, вариативность производной и др.

Вторую группу определяют характеристики, которые рассматривают временной ряд как случайный стационарный процесс: одномерная и многомерная функции распределения, одномерная и многомерная плотность вероятности случайного процесса, распределение вероятностей случайных дискретных величин, спектральная плотность.

Расширенное признаковое пространство описания исходных и преобразованных временных рядов результатов измерений во временной области представлено в таблице 3.1.

В качестве альтернативного подхода к построению классификатора характеристик может использоваться подход, основанный на типе рассчитываемых характеристик. Выделяются три основных группы характеристик – линейные характеристики, нелинейные

характеристики и другие характеристики, формируемые на основе вычисления различных мер. В составе каждой из этих групп выделяется несколько подгрупп.

Таблица 3.1 – Расширенное признаковое пространство описания исходных и преобразованных временных рядов результатов измерений

Тип преобразования	Характеристики
Исходные результаты измерений	Ошибка описания константным значением, ошибка описания линейной функцией, ошибка описания квадратичной функцией
Ранжирование значений исходных результатов измерений	Ошибка описания константным значением, ошибка описания линейной функцией, ошибка описания квадратичной функцией
Вычисление производной с применением метода конечных разностей	Ошибка описания константным значением, ошибка описания линейной функцией, ошибка описания квадратичной функцией
Вычисление нижней огибающей, верхней огибающей временного ряда результатов измерений	Ошибка описания константным значением, ошибка описания линейной функцией, ошибка описания квадратичной функцией
Вычисление вариации верхней и нижней огибающих временного ряда результатов измерений	Отклонение от нуля
Интерполяция кубическим сплайном	Ошибка интерполяции
Аппроксимация заданной функцией (степенной, показательной, логарифмической, пользовательской)	Ошибка аппроксимации
Вычисление длины кривой	Ошибка описания константным значением, ошибка описания линейной функцией, ошибка описания квадратичной функцией
Вычисление сложности кривой	Локальная, глобальная, взвешенная сложности
Вычисление вариабельности кривой	Показатели вариабельности
Вычисление характерных точек кривой	Количество максимумов, минимумов, пересечений заданного уровня
Вычисление кривизны	Максимальное и минимальное значения, медиана
Вычисление площади фигуры, ограниченной кривой и линией, соединяющей крайние точки [219]	Значение площади
Вычисление основной кривой на основе метода главных компонент [220]	Ошибка описания константным значением, ошибка описания линейной функцией, ошибка описания квадратичной функцией

При определении характеристик, относящихся к линейным, используются меры, основу которых составляет вычисление линейной корреляции, частоты, построение авто регрессионных моделей. В первую подгруппу линейных характеристик входят меры, основанные на вычислении линейной и монотонной автокорреляционных функций. В зависимости от видов шкал могут вычисляться автокорреляционные функции Пирсона, определяться ранговые корреляции Кендалла или Спирмана. При вычислении мер часто



рассматриваются различные временные задержки. К первой подгруппе также относятся характеристики, основанные на вычислении кумулятивной автокорреляции и частичной автокорреляции [221]. Подгруппа мер, вычисляемая при частотном представлении временных рядов, включает распределение энергии в частотных интервалах, медиану частоты, вычисленную на основе оценки спектральной мощности с применением стандартной периодограммы [222]–[223]. К третьей подгруппе относятся меры, формируемые на основе построения линейных моделей временных рядов: авторегрессионной модели, авторегрессионной модели скользящего среднего [224]. При вычислении мер строятся модели различных порядков. На основе построенных моделей могут быть вычислены следующие меры: абсолютный средний процент ошибки, нормализованное средне квадратичное отклонение ошибки, нормализованный средний квадрат ошибки, коэффициент корреляции.

К основным группам мер, применяемым для вычисления нелинейных характеристик, относятся меры, основанные на вычислении нелинейной корреляции, размерности и сложности временного ряда, а также построении нелинейных моделей временных рядов. К первой подгруппе нелинейных характеристик относятся две меры – бикорреляционная мера (трех точечная автокорреляция) и мера взаимной информации. Для вычисления взаимной информации применяется разбиение на равные интервалы по расстоянию и по вероятности распределения значений [225], [226]. Также как и для корреляционных мер, вычисляются меры кумулятивной бикорреляции и кумулятивной взаимной информации для различных временных задержек [227], [228]. К подгруппе мер, основанных на вычислении размерности и сложности, относятся: размерность корреляции [229], кумулятивная плотность и обратная кумулятивная плотность [230], [231]. В рассматриваемую подгруппу также включаются [232]. Мерами, оценивающими сложность, являются меры аппроксимации энтропии [233] и алгоритмической сложности [234], [235]. При вычислении мер, основанных на нелинейных моделях, рассматриваются следующие типы моделей: локальная модель среднего [236], параметры которой оцениваются с использованием метода наименьших квадратов, а также локальная линейная модель, параметры которой оцениваются на основе расчета регрессионной функции для главных компонент [237]. Для оценки моделей используются характеристики, аналогичные линейным моделям.

Меры, на основе которых определяется третья группа характеристик, подразделяются на две подгруппы: статистические меры и статистические меры для осциллирующих характеристик. К первой подгруппе относятся: среднее, медиана, стандартное отклонение, гладкость, эксцесс, первый и второй моменты временного ряда, параметры Херста [238] и корреляционные меры, которые рассчитываются с использованием метода анализа диапазонов с изменением масштаба (Rescaled range analysis) и метода анализ колебаний (Detrended actuation

analysis) [239], [240]. Вторая подгруппа включает меры для описания осциллирующих временных рядов, в частности, могут рассчитываться локальный минимум, локальный максимум, размах, продолжительность осцилляции. Многие другие меры предложены, например, в [241].

*Классификатор вариантов формализованного описания временных рядов результатов измерений.* Варианты формализованного описания определяются для стационарных, кусочно-стационарных и нестационарных временных рядов. Рассматривается два основных способа представления временных рядов – представления, адаптируемые к описываемым данным, и представления не адаптируемые к данным.

При использовании адаптируемых представлений временные ряды могут описываться следующими характеристиками: коэффициенты кусочно-линейной аппроксимации и адаптивной кусочно-константной аппроксимации; коэффициенты сингулярного разложения; символьные представления коэффициентов полиномов.

При использовании не адаптируемых представлений вычисляется следующий набор характеристик: коэффициенты вейвлет преобразования с использованием ортонормированных и биортонормированных преобразований; коэффициенты спектрального представления временных рядов, полученные с использованием дискретного Фурье-преобразования и дискретного косинусного преобразования; результаты кусочно-совокупного приближения временного ряда и другие.

*Классификатор критериев оценки данных, а также результатов работы алгоритмов.* Рассматривается три группы критериев оценки данных:

- оценки данных, формируемые исходя из сведений об источнике данных;
- оценки данных, получаемые в результате сравнения данных с историческими данными;
- оценки данных, рассчитываемые с применением специализированных процедур.

Состав критериев, относящихся к первой группе, часто представляет собой фиксированный набор рассчитываемых характеристик и допусков на их значения.

Вторая группа критериев основана на определении степени схожести обрабатываемых данных и исторических данных, полученных в аналогичных условиях.

Третья группа включает стандартные критерии оценки, определенные для алгоритмов обработки данных. При оценке качества временных рядов значений может определяться уровень шума в данных, наличие в них выбросов, трендов и смещений, пропущенных значений и другое. Оценки результатов кластерного анализа могут основываться на вычислении критериев компактности данных внутри кластера и удаленности между элементами различных кластеров. Для этого рассчитываются индекс Данна (Dunn Index) и подобные ему [242], индекс Девиса-Болдиена (Davies – Bouldinindex) [243], среднеквадратичное отклонение и коэффициент

детерминации (RMSSDT и RSIndices) [244], SD индекс [245], S\_Dbw индекс [242], в также иные, рассмотренные в [246]. Для других групп алгоритмов имеются свои критерии. Для алгоритмов ассоциации возможные критерии рассмотрены в [61], [62].

*Классификатор методов, алгоритмов и процедур обработки данных.* Классификация методов, алгоритмов и процедур проводится по нескольким признакам. Классификатор может строиться исходя из применяемого математического аппарата. Основу такого классификатора составляют стандартные классификаторы методов цифровой обработки сигналов, математической статистики и методов искусственного интеллекта. Этот классификатор часто связывается с классификатором предметных областей и классификатором задач. Взаимосвязь предметных областей, задач и алгоритмов обеспечивает возможность выбора алгоритмов при адаптивной обработке.

Другой классификатор методов, алгоритмов и процедур может быть построен исходя из основных этапов обработки. При таком подходе можно выделить группу алгоритмов, процедур и тестов, обеспечивающих получение дополнительной информации об обрабатываемых результатах измерений. К ней часто относят алгоритмы разведочного анализа, а также алгоритмы комплексной обработки и исследования данных. Это могут быть наборы тестов, включая тесты оценки качества измерений, определения типов временных рядов результатов измерений, определения набора характеристик, которыми они обладают и т.д. Основу тестов составляют различные статистические процедуры. Дополнительная информация учитывается при обработке данных с использованием алгоритмов сегментации, кластерного анализа, классификации, секвенциального анализа, поиска ассоциативных правил и других.

### **3.5.3. Программно-ориентированные модели обеспечения трансформации данных мониторинга**

На основе комплексных моделей строятся модели для программных систем. Структура программно-ориентированных моделей представлена на рисунке 3.6. Модели устанавливают взаимосвязи между объектами реального мира, областью построения моделей и областью программных систем. Рассматриваются обобщенные модели, описывающие классы моделируемых объектов, и модели конкретных объектов. Модели конкретных объектов определяются в рамках обобщенных моделей. На их основе выполняется построение моделей наблюдаемых объектов.

Подробно программно-ориентированные модели рассмотрены в работах автора [193], [247], [248]. Примеры моделей рассмотрены в [249], [250].

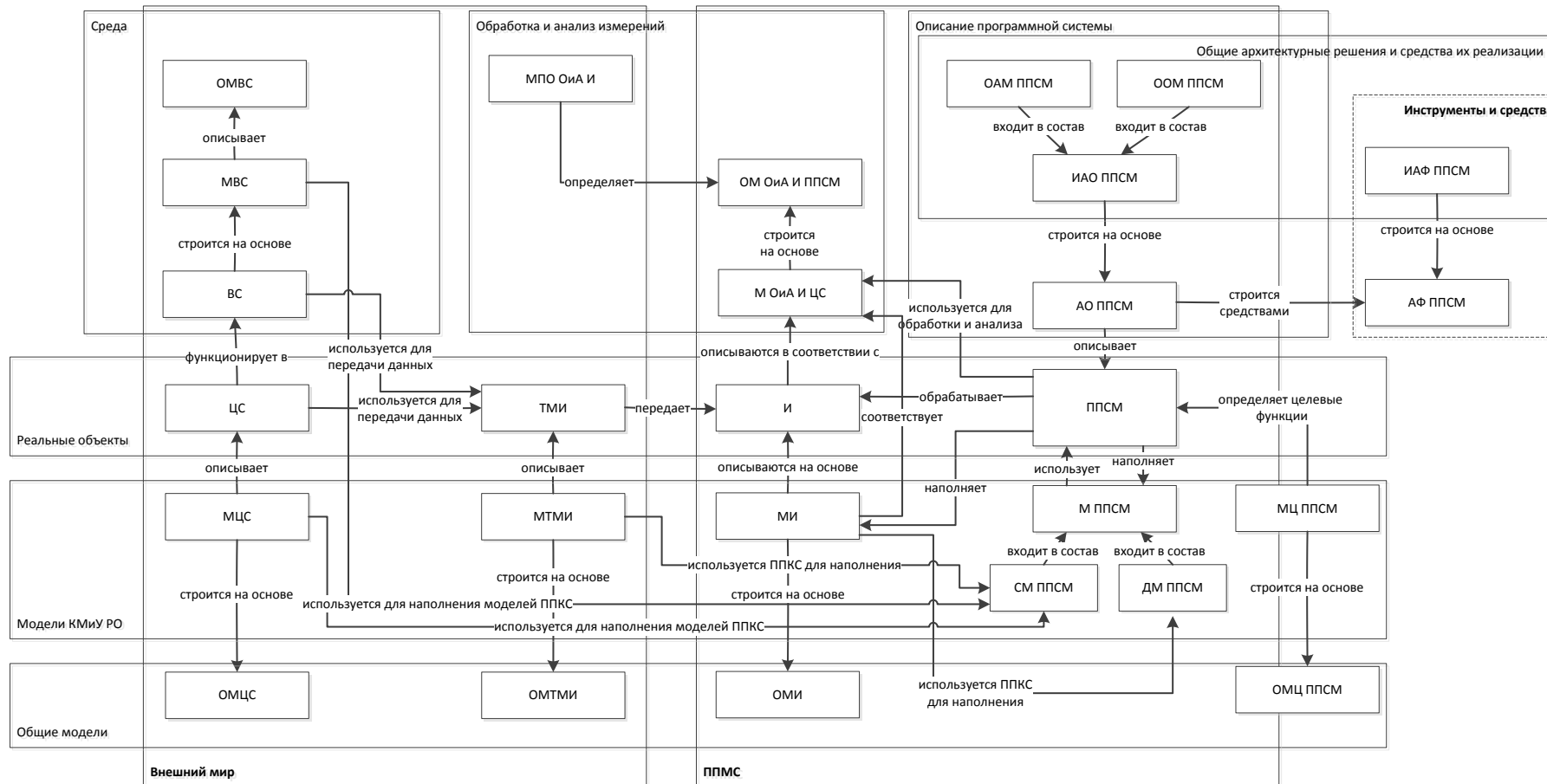


Рисунок 3.6 – Программно-ориентированная модель трансформации данных

Предложенная программно-ориентированная модель определяет следующую логику построения моделей объектов мониторинга. Наблюдаемые объекты и системы взаимосвязанных объектов (ЦС) функционируют в условиях внешней среды (ВС). Объекты и внешняя среда описываются моделью ЦС (МЦС) и моделью ВС (МВС). Для ЦС и ВС определяются обобщенные модели ОМЦС и ОМВС, которые определяют типовые параметры, характеризующие структуру и поведение объектов в условиях внешней среды. К параметрам объектов относится число элементов внутренней структуры объекта, уровни их вложенности, типы внутренних связей, число логических подсистем, возможные варианты действий и другие.

Результаты измерений параметров ЦС передаются в ППСМ с использованием системы передачи информации. В качестве систем передачи данных могут использоваться телеметрические системы (ТМИ). Средства сбора и передачи описываются моделями ТМИ (МТМИ). Для этих моделей, аналогично моделям целевых систем, определены обобщенные модели ТМИ (ОМ ТМИ). Примерами параметров моделей ТМИ являются состав измеряемых функциональных и сигнальных параметров, частота опроса датчиков, погрешности дискретизации, квантования, тарировочной шкалы, уровень помех в каналах и линиях передачи. К расчетным параметрам относятся оценки качества информации в зарегистрированных потоках измерений, оценки погрешности работы датчиков, оценки объема результатов измерений, содержащих ошибки в значениях, оценки объема потерянных кадров, оценки неточности привязки временных меток к временной шкале и др.

Результаты измерений (И), получаемые от систем передачи ТМИ, описываются в соответствии с информационными моделями представления результатов измерений (МИ). МИ строятся на основе общих моделей представления результатов измерений (О МИ).

Обработка результатов измерений осуществляется с использованием моделей, методов и алгоритмов области обработки и анализа (ОиА) данных. Для этого строятся соответствующие модели обработки (М ОиА ЦС). В общем виде процессы обработки результатов измерений описываются в виде обобщенных моделей (ОМ ОиА И). Возможности обобщенных моделей определяются возможностями модели предметной области обработки и анализа измерений (МПО ОиА И).

Информационные системы представлены в программно-ориентированной модели в виде совокупности архитектурных и онтологических моделей. Архитектурные и онтологические модели составляют архитектурное описание ППСМ (АО ППСМ). Для архитектурных и онтологических моделей определены соответствующие им обобщенные модели - ОАМ ППСМ и ООМ ППСМ.

В ППСМ данные о прикладных предметных областях и о системе размещаются в модели ППСМ (М ППСМ). Она включает две модели – статическую (СМ ППСМ) и динамическую (ДМ

ППСМ). Динамическая модель содержит информацию о текущем состоянии ЦС, ВС, ППСМ, статическая модель – значения постоянных и редко изменяющихся их параметров. Модели могут наполняться и модифицироваться на всех этапах жизненного цикла систем.

Для класса систем ППСМ определена иерархия архитектурных каркасов разработки (ИАК) (архитектурных фреймворков (ИАФ)), которые представляют собой инструментальные средства, позволяющие гибко проектировать, разрабатывать и сопровождать конечные системы. Построение и перестройка ППСМ осуществляется исходя из целей, поставленных перед системой. Цели для отдельных систем определяются в модели целей (МЦ ППСМ), для класса систем ППСМ - обобщенной модели (ОМЦ ППСМ).

На верхнем уровне система функционирует следующим образом. Формулируется задача. В соответствии с задачей формируется модель ППСМ. Затем, исходя из исходных данных и ожидаемых результатов их обработки определяется информационная модель. Для построения ИМ формируются контентно-адаптивные процессы обработки. При обработке используются МЦС и МТМИ. МЦС содержат информацию о наблюдаемых объектах, МТМИ описывают состав сырых данных, которые передаются в информационных потоках. Результаты обработки используются для построения моделей ППСМ.

### **3.6. Выводы**

1. Сформулирована задача построения многоуровневых автоматных моделей объектов мониторинга для решения которой разработан новый метод многоуровневого синтеза автоматных моделей объектов, основанный на совместном использовании индуктивного и дедуктивного подходов.

2. Разработан новый метод индуктивного синтеза моделей объектов мониторинга. Метод индуктивного синтеза обеспечивает построение многоуровневых моделей объектов на фиксированные моменты времени и на стационарных временных интервалах. При этом по имеющимся информационным потокам восстанавливаются многоуровневые связанные структуры, отражающие содержание этих потоков, выполняется их обобщение. Модели, построенные на различные моменты времени, связываются между собой на многих уровнях. В результате формируются модели, описывающие объекты на временных интервалах.

3. Разработан новый метод дедуктивного синтеза моделей объектов мониторинга. Пространство для дедуктивного синтеза формируется на основе моделей объектов, построенных с применением индуктивного синтеза. В этом пространстве становится возможным построение новых моделей объектов, способных его характеризовать на интересующие моменты времени и временные интервалы. Для этого выполняется доказательство возможности существования новых моделей.

4. Разработаны методы многоуровневого синтеза моделей процессов и программ мониторинга. Синтез процессов основан на доказательстве существования связей между целевыми моделями объектов, и моделями, синтезированными по полученным данным. При доказательстве применяется прямой нисходящий вывод. Для синтеза формальных программ применяется обратный вывод. Обратный вывод осуществляется по схеме, зеркальной прямому выводу.

5. Разработаны методы и модели многоуровневой трансформации данных. Предложенные методы и модели позволяют строить контентно-адаптивные процессы обработки данных мониторинга. В соответствии с предложенным методом вначале определяются общие последовательности трансформаций, которые затем детализируются и представляются в виде параметрических семейств процессов обработки. Модель трансформаций определяет способы представления последовательностей трансформаций и результатов их детализации на различных уровнях. В частности, семейства процессов представляются в виде многоуровневых горизонтально-вертикальных структур.

6. Разработаны методы и модели обеспечения многоуровневой трансформации данных мониторинга, позволяющие строить конечные процессы обработки данных в соответствии с предложенным методом трансформации. Предложена обобщенная модель, которая раскрывается в виде связанных информационной, функциональной, процессной и сервисной моделей. Правила их построения определяются в управляющей модели.

В целом, в третьей главе разработаны новые: методы многоуровневого автоматического синтеза автоматных моделей объектов мониторинга; методы многоуровневого автоматического синтеза автоматных моделей процессов и программ мониторинга для построения моделей объектов; методы и модели многоуровневой адаптивной трансформации данных мониторинга для построения моделей объектов.

## **Глава 4. Методы и модели разработки проблемно- и предметно-ориентированных систем построения моделей объектов по данным мониторинга**

### **4.1. Метод разработки проблемно- и предметно- ориентированных систем построения моделей объектов по данным мониторинга**

Проблемно- и предметно- ориентированные системы построения моделей объектов по данным мониторинга (ППСМ) определяют новый класс систем построения моделей объектов. Это системы, предназначенные для решения различных задач и относящиеся к различным предметным областям, но использующие похожие решения.

Отличительной особенностью ППМС является наличие у них собственных моделей, моделей наблюдаемых объектов и моделей внешней среды, способность работать с этими моделями. При работе с моделями в системах ППМС широко используются методы и средства ИИ. Процессы, реализуемые в системах, являются адаптивными. Возможности адаптации предусматривают как настройку параметров процессов, так и генерацию новых процессов. Адаптация осуществляется при изменении состава данных, состава решаемых задач, условий обработки данных. В условиях, когда возможностей систем оказывается недостаточно для решения поставленных задач, системы перестраиваются, т.е. можно говорить о способности систем не только к адаптации, но и к самоадаптации [251]–[260].

Таким образом, ППМС – это подкласс адаптивных интеллектуальных систем, обладающих способностями к перестройке или, иначе, самоадаптивных интеллектуальных систем.

Область применения ППМС определяется в терминах предметных областей и решаемых задач. Область, ограниченная одной или несколькими предметными областями и / или решаемыми в них задачами, носит название домена.

Метод разработки проблемно- и предметно- ориентированных систем построения моделей объектов основан на следующих основных положениях:

1. ППСМ имеют гибкую архитектуру. Реализуется модельный подход к описанию систем, предусматривающий представление систем в виде совокупности связанных моделей, преобразование которых приводит к преобразованию самих систем.
2. ППСМ имеют иерархическую структуру, включающую многие подклассы систем, на основе которых формируются конечные системы.
3. Разработка ППСМ предполагает использование гибких методологий при проектировании, создании, применении и сопровождении систем.
4. Разработка ППСМ ориентирована на использование готовых высокоуровневых элементов, включая программные платформы и т.д.



5. Разработка ППСМ поддерживается инструментальными и программными средствами, включающими архитектурные каркасы (архитектурные фреймворки) и их расширения.

Подробное описание ППСМ и предлагаемых методов их разработки приведено в монографии [261], а также в статьях [262]–[266].

Основные понятия, используемые при разработке ППСМ, приведены на рисунке 4.1. К ним относятся ядро ППСМ, архитектурный каркас разработки ППСМ, инструментарий разработки ППСМ.

Системы строятся на основе ядра ППСМ. Ядро представляет собой интеграционную платформу. В нем содержится информация о компонентах, которые могут включаться в состав ППСМ. К таким компонентам относятся наборы базовых сервисов, в число которых входят функциональные сервисы и сервисы-движки, модели, алгоритмы, библиотеки, онтологии. Возможна интеграция высокоуровневых компонентов, включая готовые технологии, платформы.

Одной из ключевых составляющих ядра являются архитектурные знания. В целом, интерпретация термина «архитектурное знание» совпадает с предложенным в [267], но в данной работе он используется в более узком смысле - «явное знание, содержащееся в моделях систем». Архитектурные знания включают знания о типовых архитектурных и структурных решениях, процессах проектирования и другие.

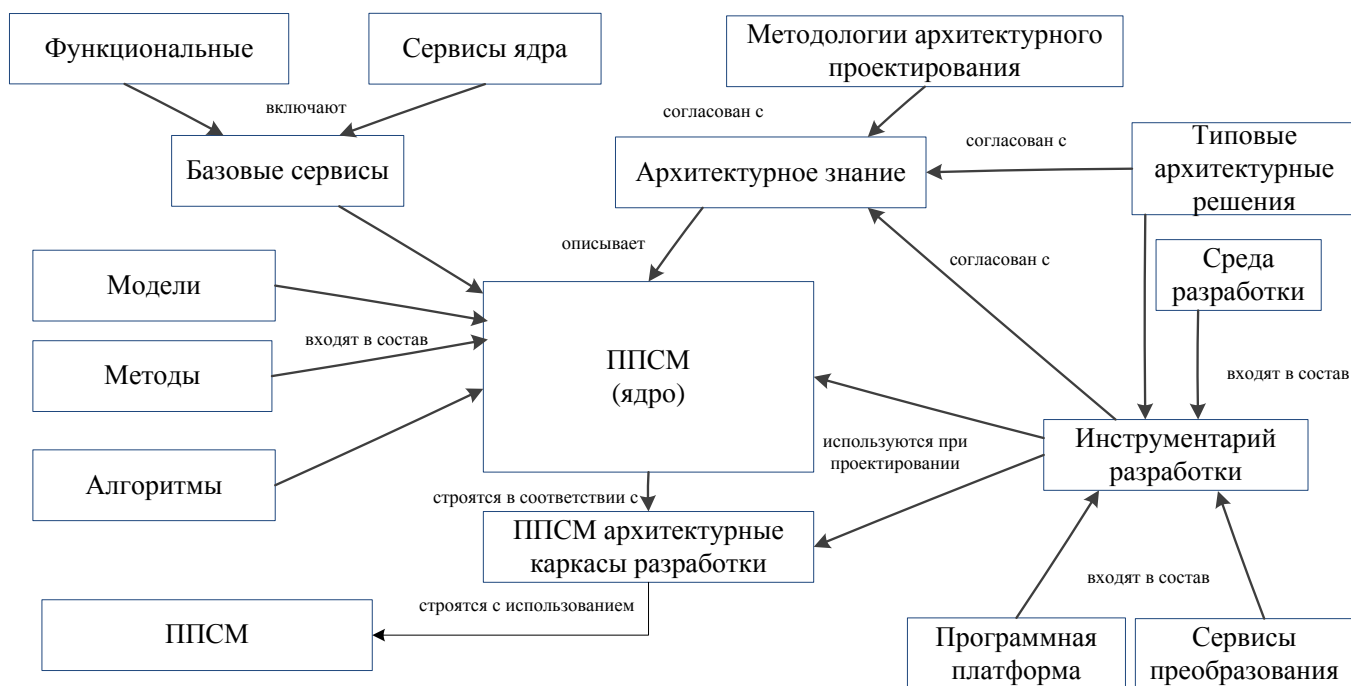


Рисунок 4.1 – Основные понятия для класса систем ППСМ и соотношения между ними

ППСМ разрабатываются с использованием архитектурных каркасов разработки. Такие каркасы также носят названия архитектурных фреймворков разработки (Architectural Framework, AF). Архитектурный фреймворк (АФ) - это инструмент проектирования ППСМ, который может рассматриваться, с одной стороны, как мета-архитектура, на основе которой можно создавать

целевые архитектуры, а, с другой стороны, данный АФ можно рассматривать как самостоятельную систему, реализующую гибридную архитектуру и предназначенную для использования в качестве генератора архитектур. При этом разработка ППСМ сводится к адаптации типовых решений к особенностям домена, для которого система строится.

Процессы разработки ППСМ поддержаны инструментально. Инструментарий разработки включает средства для работы с моделями систем и средства для построения систем по их моделям. Разработка ППСМ ведется в специализированной среде разработки (Builder, Билдер). Для работы с моделями предусмотрены сервисы преобразования моделей. Архитектурные и структурные решения (АСР) строятся на основе типовых. Программная платформа предоставляет программные реализации компонентов, включаемые в формируемые модели систем.

Разрабатываемые системы обладают гибкой (Agile) архитектурой, в которой заложена возможность адаптации систем на разных этапах жизненного цикла: проектирования, разработки, сопровождения. Необходимость в адаптации систем возникает при изменении предъявляемых к ним требований. В зависимости от этапа жизненного цикла достигаемые цели, масштаб и сложность адаптации систем могут существенно отличаться. На этапе проектирования систем архитектура создается “с нуля” или на основе архитектур существующих в предметной области систем. При сопровождении в большинстве случаев требуется внесение существенных изменений в логику работы и структуры систем. На этапе использования систем предусматривается возможность их настройки. Настройка может состоять в локальной переконфигурации логики работы систем, подключении отдельных программных модулей.

Значительный положительный эффект при разработке ППСМ достигается за счет совместного использования уже существующих технологий [268]. На сегодняшний день интеграция технологий осуществляется на интуитивном уровне и возможности этого подхода практически не используются на практике. Для ППСМ в качестве базовых выступают следующие технологии: технологии доменно-ориентированного проектирования, технологии построения семейств и линеек систем, технологии построения гибких архитектур [251]–[253], технологии адаптации систем [254], интеллектуальные технологии. Расширенный список технологий может включать: технологии работы с сетями датчиков [269], технологии слияния данных [270], технологии построения онтологических моделей [271], технологии проектирования сервисно-ориентированных архитектур (COA) [272], технологии проектирования на моделях (Model Driven Engineering, MDE) [273], Grid (грид) технологии [274].

В свою очередь, ППСМ может входить в качестве составной части в другие технологии: технологии Интернета вещей; технологии, применяемые при создании умных (Smart) систем и другие.

## 4.2. Модели гибкой архитектуры ППСМ

### 4.2.1. Общая модель гибкой архитектуры ППСМ

ППСМ имеют гибкую архитектуру, которая описывается на модельном уровне (рисунок 4.2). Представленная общая архитектурная модель расширяет и развивает модель, определенную стандартом ISO/IEC/IEEE 42010:2011.

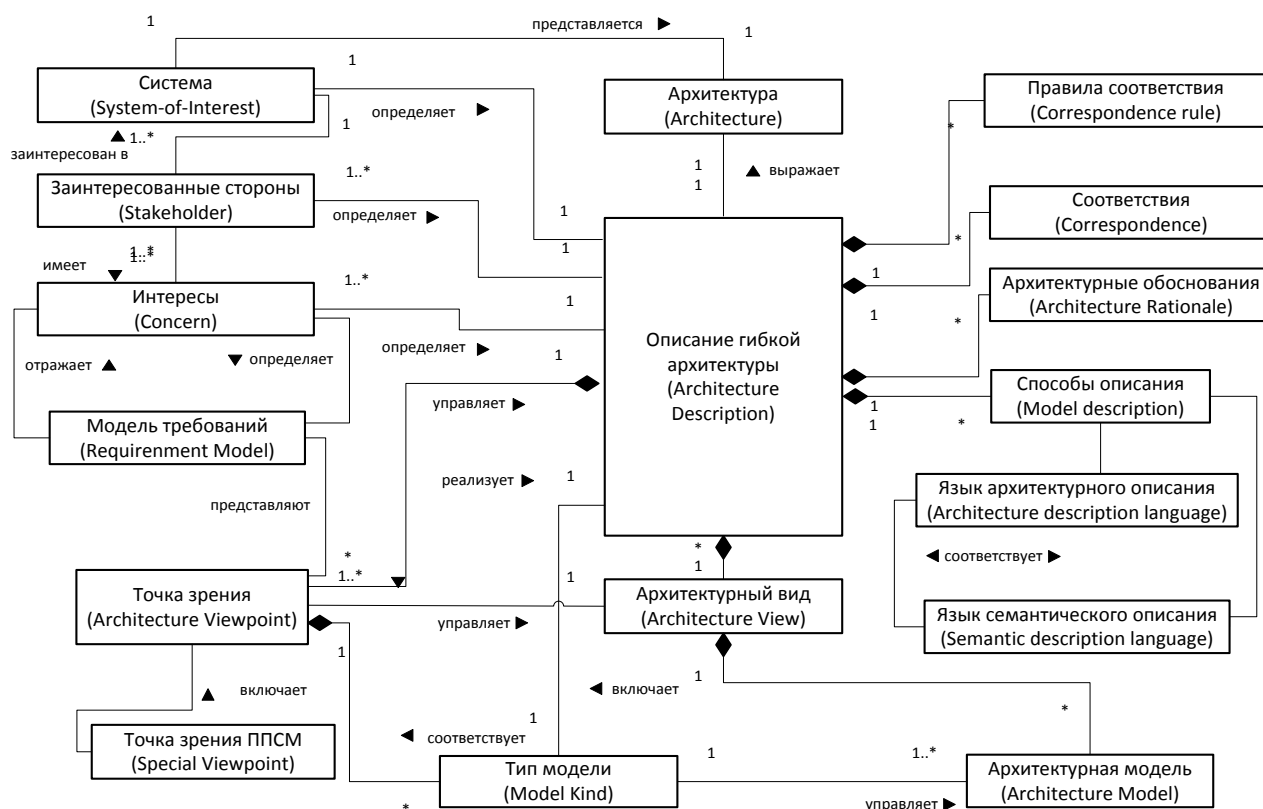


Рисунок 4.2 – Общая модель гибкой архитектуры ППСМ

В оригинальном документе используются следующие отношения: заинтересован в (has interest in), имеет (has), отражает (frames), определяет (identifies), представляется (exhibits), выражает (expresses), управляет (governs), включает (addresses).

Следует заметить, что стандарт [275] четко различает понятия архитектура (Architecture) и архитектурное описание (Architecture description), при этом в качестве предмета стандартизации выступает архитектурное описание. Стандарт определяет минимум требований к архитектурному описанию. Архитектурное описание – это некоторый артефакт или совокупность артефактов, а архитектура – это абстракция, включающая в себя концепции и свойства. Стандарт [275] не определяет процессы, используемые для проектирования архитектуры, и не регламентирует состав используемых моделей и нотаций.

Архитектурное описание сопровождает систему на протяжении всего жизненного цикла и может использоваться для решения следующих основных задач: как инструмент проектирования системы; как язык общения заинтересованных сторон; для оценки альтернативных вариантов

реализации; для составления спецификаций семейств и линеек систем; для составления документации на системы; в качестве исходных данных при разработке и сопровождении систем; при проведении работ, связанных с переходом на новые архитектуры и т.д.

Заинтересованные стороны (Stakeholders) имеют интересы относительно некоторой системы (System-of-interest), несколько заинтересованных сторон могут иметь общий интерес (Concern). В соответствии с [275] определяются следующие типовые группы заинтересованных сторон: пользователи (Users), операторы (Operators), покупатели (Acquirers), владельцы (Owners), поставщики (Suppliers), разработчики (Developers), наладчики, изготовители, специалисты по эксплуатации (Maintainers). Интересы появляются на протяжении всего жизненного цикла системы. Интересы могут быть обозначены в разных формах: в форме целей, ожиданий, требований, ограничений, показателей качества, рисков и т.д.

Точка зрения (Architecture viewpoint) включает один или более интересов, при этом каждый из интересов может входить в несколько точек зрения. Точка зрения определяет вид (Architecture view) в соответствии с интересом. Каждый интерес должен быть ассоциирован, по крайней мере, с одной точкой зрения. Для каждой точки зрения архитектурное описание должно включать ровно один вид, а каждый архитектурный вид должен быть связан с определяющей его точкой зрения. Точка зрения содержит соглашение о видах. Соглашение о видах может включать языки, нотации, виды используемых моделей, правила проектирования, методы моделирования и др.

Архитектурный вид связывается с одной или несколькими моделями, каждая из которых имеет идентификатор версии. Каждая архитектурная модель (Architecture model) принадлежит определенному типу и использует соглашения о способах моделирования, которые определяются типом модели (Model kind). Для описания типов моделей могут использоваться метамодели. Должны определяться преобразования типа модели в конкретный экземпляр.

Архитектурное описание определяет отношения между всеми рассмотренными элементами. Отношения между элементами определяются в терминах соответствия (Correspondence). Отношения представляются в виде правил соответствия (Correspondence rule). Для объяснения того, почему приняты те или иные архитектурные решения используются архитектурные обоснования (Architecture rationale).

Для предлагаемой модели гибкой архитектуры ППСМ рассматривается расширенный набор точек зрения, которые определяются как точки зрения ППСМ (Special Viewpoint). При описании архитектуры предусмотрено совместное использование архитектурных и онтологических описаний [218], [276]–[279]. Для этого в модели определяются языки архитектурного описания (Architecture Description Language) и семантического описания (Semantic Description Language).

#### 4.2.2. Архитектурные модели систем ППСМ

Архитектурная модель представляет собой архитектурное описание ППСМ в рамках стандарта [275]. Модель ППСМ включает в себя следующие точки зрения (Viewpoint, ТчЗ) (таблица 4.1): обобщенная ТчЗ, системная ТчЗ, функциональная ТчЗ, модельная ТчЗ, информационная ТчЗ, процессно-сервисная ТчЗ, объектная ТчЗ, проектная ТчЗ, ТчЗ, учитывающая стандарты.

Предлагаемую модель можно позиционировать как полицентрическую. В зависимости от точки зрения, интерес могут представлять парадигмы: датацентрическая (Data Centric), процессноцентрическая (Process centric). Возможно рассмотрение сетецентрической (Net centric) парадигмы. Подробное описание этих парадигм можно найти в [280].

Таблица 4.1 – Состав и структура точек зрения архитектурного описания ППСМ

ТчЗ, учитывающая стандарты	Обобщенная ТчЗ	Системная ТчЗ	Проектная ТчЗ
		Функциональная ТчЗ	
		Модельная ТчЗ	
		Информационная ТчЗ	
		Процессно-сервисная ТчЗ	
		Объектная ТчЗ	

Краткое описание ТчЗ приведено в таблице 4.2.

При определении точек зрения для архитектурных моделей ППСМ учитывалось:

1. Для систем ППСМ информационная ТчЗ является одной из основных, поскольку системы ориентированы на обработку данных.

2. Динамика функционирования ППСМ описывается в рамках процессной точки зрения. При этом процессы чаще всего описываются на уровне сервисов. Как будет показано ниже, СОА является одним из базовых решений при построении ППСМ. В рамках систем ППСМ процессы определяются как последовательность обращений к сервисам, каждый процесс при обращении к нему рассматривается как сервис. Поэтому разделять процессы и сервисы представляется нецелесообразным.

3. На сегодняшний день при проектировании ИС широко используется объектно-ориентированный подход, который хорошо поддержан инструментальными средствами. Следует отметить, что в настоящее время достаточно хорошо проработаны вопросы прямого и обратного перехода между ОО и онтологическими представлениями [281].

Прочие ТчЗ являются достаточно традиционными и присутствуют практически во всех архитектурных фреймворках.

Таблица 4.2 – Точки зрения архитектурного описания ППСМ

№	Точка зрения	Обозначение	Описание точки зрения
1	Обобщенная точка зрения	All Viewpoint, AV	Интегрирует все точки зрения и образует архитектурный контекст для других точек зрения
2	Системная точка зрения	Systems Viewpoint, SV	Рассматривает систему как совокупность взаимодействующих подсистем, рассматривает способы взаимодействия подсистем, включает структурное описание системы
3	Функциональная точка зрения	Capability Viewpoint, CV	Определяет реализуемый функционал
4	Модельная точка зрения	Model Viewpoint, ModV	Описывает систему в терминах используемых моделей
5	Объектная точка зрения	Object Viewpoint, ObjV	Рассматривает систему в терминах ОО моделей
6	Информационная точка зрения	Information Viewpoint, IV	Описывает информационные структуры для представления исходных данных и результатов их обработки
7	Процессно-сервисная точка зрения	Service-Process, PR	Описывает функционирование системы в терминах процессов и сервисов, реализуемых на разных уровнях
8	Проектная точка зрения	Project Viewpoint, PV	Рассматривает систему с точки зрения требуемых характеристик и возможностей, а также с точки зрения процесса проектирования
9	Точка зрения, учитывающая стандарты	Standards Viewpoint, StdV	Учитывает, в частности, действующие технические стандарты, методики, руководства, ограничения и т.п.

В качестве основных заинтересованных сторон (Stakeholders) выступают: пользователи, администраторы, разработчики, владельцы и продавцы системы.

Подробное описание отдельных точек зрения приведено в [282].

При построении архитектурных моделей ППСМ в качестве базового может использоваться подход, известный как проектирование на моделях (Model Driving Engineering, MDE). Идея MDE подхода состоит в построении стека моделей [283].

Основными для MDE являются следующие понятия: модель (model, M), метамодель (Metamodel, MM), связывание моделей (Model Weaving, MW), преобразование моделей (Model Transformation, MT).

При построении стека моделей ППСМ используется определение модели, предложенное в [284]. Стек моделей ППСМ содержит пять уровней M0-M4 (таблица 4.3).

Уровень M0 представляет собой описание конечной системы, уровень M1 соответствует модели конечной системы.

На уровне M2 размещаются метамодели, которые определяются для предметных областей и решаемых в них задач. Они носят название доменных моделей. Эти модели можно считать ключевыми моделями для ППСМ.

Уровень М3 – это уровень метамodelей, определяющих класс систем ППСМ. Каждая модель конечной системы должна соответствовать некоторой метамodelи, которая определяет ее структуру и семантику в более общем виде. Метамodelь включает описание ключевых сущностей, отношений и ограничений языка моделирования.

Уровень М4 - это уровень метамodelей для метамodelей, которые называют мета-метамodelи. Поскольку уровень мета-метамodelей является самым высоким уровнем, то мета-метамodelи соответствуют только сами себе.

Таблица 4.3 – Стек архитектурных моделей ППСМ

Уровень	Описание
Уровень М4 (верхний уровень)	Уровень метамodelей для метамodelей
Уровень М3	Метамodelи класса систем
Уровню М2	Модели доменных систем
Уровню М1	Модели конечных систем
Уровень М0 (нижний уровень)	Конечная система

Ограничений на вложенность моделей в рамках каждого из уровней не налагается. Как правило, модели, относящиеся к одному уровню, отличаются степенью их детализации.

Для преобразования моделей систем могут применяться операции трансформации и связывания. Процесс трансформации моделей – это процесс, который определяет, каким образом можно получить целевую модель из одной или нескольких исходных моделей. Модель процесса трансформации представлена на рисунке 4.3. Пусть модель Ма, соответствующая метамodelи ММа, трансформируется в модель Мб, которая соответствует метамodelи ММб. Само преобразование можно определить как МТ. МТ также представляет собой модель, которая соответствует метамodelи ММТ. ММТ, в свою очередь, является моделью и должна соответствовать мета-метамodelи, например, MOF (Meta-Object Facility) модели.

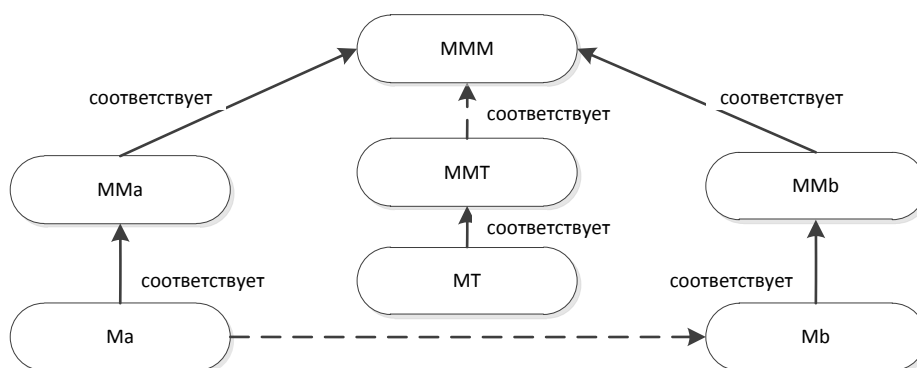


Рисунок 4.3 – Модель трансформации

Модели могут трансформироваться либо «по горизонтали», либо «по вертикали». В первом случае трансформируемая модель преобразовывается в модель, которая принадлежит одному с

ней уровню. Во втором случае трансформируемая модель является моделью более высокого уровня.

С точки зрения типов трансформируемых моделей можно выделить следующие основные варианты [285]:

- а) трансформация типа «формальная модель – формальная модель»;
- б) описание, имеющее формализованную синтаксическую структуру – формальная модель;
- в) текстовое описание – описание, имеющее формализованную синтаксическую структуру;
- г) текстовое описание – формальная модель;
- д) формальная модель – программный код.

Таблица 4.4 – Механизмы трансформации архитектурных моделей

№	Механизм трансформации модели	Описание механизма трансформации модели
1	Прямое или низкоуровневое преобразование	Предполагает использование внутреннего представления моделей и API для работы с моделями. Может реализовываться в виде объектно-ориентированного фреймворка. Требуется минимальных затрат с точки зрения использования
2	Структурное преобразование	Предполагает использование двухэтапного подхода. На первом этапе создается целевая модель, а на втором этапе осуществляется ее наполнение, при этом пользователь задает правила трансформации
3	Операционные преобразования	Идентичны прямым преобразованиям за исключением того, что предполагают использование метамodelей
4	Преобразования на основе шаблонов	Основываются на использовании шаблонов моделей, которые представляют собой, как правило, модели, содержащие метакод. Метакод инкапсулирует особенности конкретного экземпляра шаблона и представляется в виде OCL выражений
5	Реляционный подход	Основывается на определении отношений между элементами исходной и целевой моделей. Отношения могут определяться посредством описания отображений, логических формул и т.п. Основным достоинством данного подхода является наличие формальной составляющей, однако если требуется использовать сложные законы, описывающие соответствия, то использование подхода становится затруднительным
6	Подход на основе преобразования графов	Основывается на исследованиях в области теории графов. Однако, несмотря на определенные достижения, общая теория отсутствует
7	XSLT преобразования	Подход к преобразованию XML документов, основанный на использовании движка и сценариев. На вход движка подается исходный документ и сценарий (скрипт, содержащий правила преобразования). Механизм широко используется для преобразования документов
8	Преобразования с использованием онтологий	Основывается на работе с онтологиями. Онтология используется как средство определения эквивалентности понятий
9	Гибридный подход	Представляет собой комбинацию нескольких описанных выше подходов. Подход реализуется, в частности, в определенном Object Management Group стандартном наборе языков, известном как QVT (Query/View/Transformation) модель



Можно определить следующие базовые механизмы трансформации моделей: прямые (Direct manipulation), структурные (Structure Driven), операционные (Operational) преобразования, преобразования с использованием шаблонов (Template-based), реляционные (Relational), преобразования графов (Graph Transformation) [286], XSLT-преобразования (XSLT-based), гибридные (Hybrid) преобразования. Описание механизмов трансформации моделей приведено в таблице 4.4.

Для связывания моделей разных типов предназначен специальный тип моделей - модели связывания. Основное назначение этих моделей состоит в установлении соответствия между отдельными элементами других моделей (низкоуровневое связывание).

Модель связывания может быть использована для связывания любых элементов разных метамodelей, таких как метаклассы, атрибуты, ссылки, типы данных. На рисунке 4.4 модель связывания WM используется для установления соответствия между элементами метамodelей MMa и MMb. При этом сама WM должна соответствовать метамодели связывания WMM.

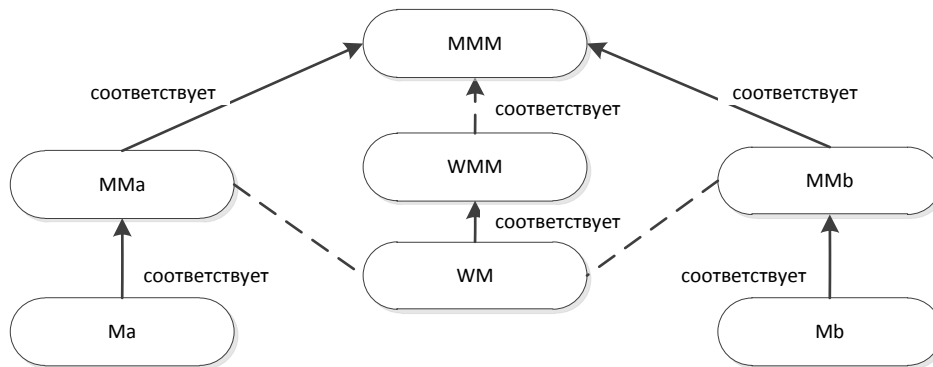


Рисунок 4.4 – Модель связывания

При необходимости использования для описания архитектуры системы множества метамodelей вводится понятие более высокого уровня, в качестве которого выступает понятие мегамодели [287]. Модели для мегамоделирования предложены в [287]. В [287] определены модели следующих типов: терминальные модели (Terminal Models), которым соответствуют реальные системы и некоторые метамodelи; метамodelи (Meta Models), которые определяют доменные модели; мета метамodelи (Meta Metamodels), которые описывают модели верхнего уровня и соответствуют сами себе. Метамodelи и мета-метамodelи выступают в качестве эталонных моделей (Reference Models), поскольку они представляют собой своего рода эталон для моделей, которые должны им соответствовать. Терминальные модели подразделяются на модели трансформации (Transformation Models), модели связывания (Weaving Models) и мегамodelи (Mega Models). Модели трансформации и связывания определяют процессы преобразования моделей и определения отношений между моделями, а мегамodelи предназначены для поддержки процесса мегамоделирования. Терминальная модель не является абстрактным метаклассом и

может использоваться для порождения конкретных экземпляров и поэтому должна соответствовать одной метамодели.

#### 4.2.3. Онтологические модели ППСМ

Онтологические модели – это основная форма представления и хранения архитектурного знания. За счет применения онтологических моделей во многом обеспечивается гибкость ППСМ.

Онтологические модели могут применяться в процессе создания, сопровождения и (или) модификации систем. Такие системы относят к системам, проектирование которых осуществляется на основе использования онтологий (Ontology-driven Development of the Information System). Если онтологии используются только в процессе функционирования (Run time) систем, то такие системы рассматриваются как системы, управляемые онтологиями (Ontology-driven Information system) [271].

Перспективность применения онтологических моделей обосновывается тем, что создание и использование таких моделей позволяет преодолеть важные недостатки существующих подходов, применяемых при проектировании и сопровождении информационных систем. К основным недостаткам существующих подходов относятся следующие:

1. Широко используемые методы проектирования ориентированы, в первую очередь, на возможно более быстрое получение работоспособного кода, что, естественно, не самым положительным образом влияет на стоимость сопровождения и возможность повторного использования кода.

2. Применяемые средства архитектурного описания информационных систем, к которым относятся MDA и UML, не предоставляют удобных инструментов для описания множества альтернативных решений.

3. Время жизни конкретного изделия и время жизни отдельной системы становятся сопоставимыми. Достаточно часто возникает ситуация, когда каждая новая система, входящая в состав семейства или линейки, или дополнительные функциональные модули реализуются уже на новых платформах.

Описанную выше ситуацию можно улучшить, если на этапе проектирования систем использовать опыт, накопленный в ходе предыдущих разработок. Это обеспечивается посредством применения механизмов работы со знаниями. На сегодняшний день стандартом для представления знаний о системах являются онтологии [288].

Следует отметить, что метамодели и онтологии имеют много общего, в частности, предполагают использование одних и тех же языков, но их назначение и характеристики существенно различаются. В соответствии с [289] разница состоит в том, что онтология описывает

проблемы, принадлежащие некоторой предметной области, а метамодель описывает множество вариантов решения проблемы.

В связи с этим наиболее прагматичным представляется подход, предполагающий использование двух параллельных описаний: онтологического и MDA описания. Для описания архитектурных моделей обычно применяется язык UML, а онтологических – язык OWL [289]. Таким образом, в UML – OWL описаниях систем UML является средством представления, а онтологии – средством описания доменов и множества возможных архитектурных и структурных решений.

Следует отметить, что вопросы прямого и обратного преобразования между двумя упомянутыми описаниями достаточно давно и активно прорабатываются, при этом уже доступны такие средства как: Metadata Interchange (XMI), Ontology Definition Metamodel (ODM), UML Profile (OUP), которые позволяют, в том числе, создавать онтологии средствами UML.

С учетом необходимости и возможности использования онтологических описаний при проектировании систем, процесс проектирования ППСМ организуется в соответствии со следующими принципами.

1. Процесс проектирования оптимизируется применительно к множеству семейств архитектур, состав которых определяется предложенной иерархией моделей.
2. На всех основных этапах жизненного цикла для описания систем используется два параллельных взаимодополняющих описания – MDA-UML описание и онтологическое описание.
3. Процесс проектирования описывается в терминах трех жизненных циклов: жизненный цикл проблемного домена для которого разрабатывается система, жизненный цикл семейства систем и жизненный цикл отдельной системы.

Можно выделить следующие основные аспекты использования онтологий на этапе проектирования:

- использование онтологий, описывающих предметную область, при анализе и постановке задач проектирования;
- формирование структур баз данных на основе онтологий;
- формирование архитектурных описаний пользовательских интерфейсов на основе конвертации информации, содержащейся в онтологиях;
- формирование архитектурного описания системы и (или) ее компонентов на основе конвертации информации, содержащейся в онтологиях;
- формирование на основе онтологий системы правил, определяющих поведение объектов предметной области;
- использование онтологий в качестве словаря понятий.

Известно достаточно много предложений по использованию моделей, которые определяют, по крайней мере, частично онтологии, ориентированные на поддержку процесса проектирования информационных систем. Наиболее интересен подход, описанный в [289]. В соответствии с данным подходом, онтология, ориентированная на поддержку процесса проектирования, состоит из трех подонтологий. К ним относятся:

- онтология метамodelей (Metamodel Domain), которая содержит набор базовых концепций, определяющих принципы построения программных систем;
- онтология методов (Method Domain), описывающая методы получения экземпляров из классов, представленных в онтологии метамodelей;
- онтология возможных реализаций (Endeavour Domain), которая содержит описания возможных реализаций классов, описанных в онтологии методов.

Этап разработки систем сводится к определению состава и способа использования онтологий в процессе функционирования систем.

Можно выделить два подхода к использованию онтологий в процессе функционирования ИС: онтологии используются непосредственно для преобразования информации, онтологии используются для управления процессом переработки информации. В рамках первого подхода используется большое число паттернов архитектурного уровня. В качестве основных архитектурных паттернов можно определить следующие [290]:

- использование онтологии в качестве хранилища данных (метаданных);
- формирование запросов к онтологии (базе знаний) и (или) выполнение логического вывода, использование результатов при принятии решений;
- по онтологии формируется пользовательский интерфейс;
- содержимое онтологии используется для настройки параметров прикладного программного обеспечения.

В рамках второго подхода можно выделить следующие основные варианты использования онтологий: использование содержимого онтологий для формирования бизнес-правил, использование онтологий в процессе функционирования динамических ассемблеров, использование стандартных онтологических описаний, таких как OWL-S для обеспечения функционирования семантических Web (Grid) -сервисов, использование онтологий в процессе функционирования мультиагентных систем [291].

После своего создания система переходит в фазу использования и сопровождения, которая может длиться в течение нескольких лет. При сопровождении выполняется исправление выявленных ошибок, изменение и (или) расширение функциональности системы с целью ее адаптации к изменяющимся требованиям. Поскольку фаза сопровождения растягивается на несколько лет, то не только знания, накопленные в процессе разработки системы, но даже

документация очень часто оказывается неполной, а иногда и утраченной. На практике оказывается, что при решении задачи модификации системы от 40 до 60% времени уходит на то, чтобы специалисты, занимающиеся модификацией ИС, смогли понять, как работает система [292].

На этапе сопровождения онтология может наполняться следующей информацией:

- об архитектурных решениях, структуре и организации системы, способе ее реализации, предусмотренных типовых конфигурациях;
- о текущей структуре, организации и конфигурации системы, составе используемых в системе программных компонентов, текущем техническом состоянии системы;
- об истории эксплуатации системы, включающей описание состава пользователей и решенных задач на различных этапах ее жизненного цикла.

Задача модификации системы может рассматриваться как некоторый специальный тип проекта, таким образом, онтология, которая может обеспечить эффективный процесс модификации системы, должна представлять собой некоторый «срез» онтологии проектирования.

На всех этапах жизненного цикла ППСМ используются следующие механизмы работы со знаниями, представленными в форме онтологий:

- извлечение знаний из данных (Data Mining);
- извлечение знаний о ходе выполнения процессов из лог файлов (Process Mining);
- добавление знаний в онтологии (обучение);
- выполнение логического вывода в рамках систем поддержки принятия решений;
- преобразование знаний, представленных в форме онтологий (редактирование, слияние онтологий, выделение профилей);
- преобразование знаний в правила;
- получение описаний из онтологий, включая UML и ER представления.

### **4.3. Метод гибкого проектирования и сопровождения ППСМ**

При проектировании ППСМ используется доменно-ориентированный подход к проектированию информационных систем. Доменно-ориентированное проектирование является действенным средством повышения эффективности функционирования ИС и уменьшения сроков и стоимости проектирования и, соответственно, суммарной стоимости владения.

В настоящей работе доменная ориентация понимается в смысле, раскрытом в [293]–[295]. На основе понятия доменной ориентации развивается подход, основанный на построении доменов решений, т.е. классов информационных систем, относящихся к разным предметным областям и позволяющим решать разные задачи, но использующих похожие решения. Предлагаемый подход к построению доменов решений дополняет и развивает существующие решения для проектирования систем [296].

В настоящее время хорошо проработаны два подхода к проектированию. Один из них направлен на разработку принципов организации и проектирования ИС, относящихся к конкретному домену задач (банковские, производственные системы и т. д.) [297]. Такие системы имеют узкую область применения. Другой подход ориентирован на создание и использование доменно-ориентированных языков проектирования (Domain Specific Languages, DSL). Однако, данный подход хорошо работает только для крупных доменов, а для небольших доменов создание собственных DSL оказывается экономически необоснованным.

Несмотря на то, что подход к построению доменов решений позволяет преодолеть перечисленные недостатки, на практике он используется достаточно ограничено. Основная проблема, связанная с применением данного подхода, состоит в отсутствии методических и инструментальных средств, позволяющих эффективно адаптировать решения доменного уровня применительно к конкретным предметным доменам.

Предлагаемый метод гибкого проектирования и сопровождения ППСМ предусматривает построение иерархии фреймворков, предназначенных для проектирования ППСМ различного уровня: систем доменного уровня, семейств и линеек систем, конечных систем. При проектировании систем могут использоваться различные архитектурные стили. Непосредственно для ППСМ предложена новая трехслойная архитектура, основанная на комбинированном применении веб сервисов и агентов. В результате проектирования строятся модели систем. При этом модели систем более низких уровней строятся за счет трансформации и связывания моделей более высоких уровней. Реализация таких моделей обеспечивается за счет интеграции готовых программных компонент. Построенные конечные системы конфигурируются. Вопросы гибкой конфигурации систем рассмотрены в статьях [98]–[99].

Таким образом, новый метод определяет единую гибкую методологию разработки программных систем.

#### **4.3.1. Процесс гибкого проектирования и сопровождения ППСМ**

Проектирование ППСМ осуществляется в три этапа (таблица 4.5).

На первом этапе выполняется проектирование систем для некоторого домена. Результаты представляются в виде платформенно-независимой эталонной модели и доменной онтологии. При таком проектировании могут использоваться различные мета модели.

На втором этапе проектируются семейства систем. Для каждого семейства определяются эталонные архитектуры и онтологические описания архитектурных и структурных решений. Описания относятся к конкретным архитектурам, например, сервисно-ориентированной архитектуре, грид, мультиагентным архитектурам. Основу этих моделей составляют доменные модели.

На последнем, третьем этапе, проектируются конечные системы. К артефактам третьего уровня относятся UML диаграммы и онтологии низкого уровня, ориентированные на использование при решении задач конкретной системой. Онтология задач может описываться в терминах OWL. Онтология методов их решения может включать описание веб-сервисов в терминах WSDL, семантических веб-сервисов в терминах OWL-S, грид сервисов, возможных вариантов поведения агентов и т.д.

Таблица 4.5 – Этапы проектирования ППСМ

Архитектурное проектирование		Онтологическое проектирование	
Процесс	Артефакт	Процесс	Артефакт
1	Доменное проектирование	Проектирование доменной онтологии	Доменная онтология
2	Проектирование семейства систем	Проектирование онтологий уровня задач	Онтологии уровня задач
3	Проектирование конечной системы	Проектирование онтологий уровня приложений	Онтологии, используемые в процессе функционирования, поддержки и модификации систем

Представленная иерархия моделей проектирования ППСМ обладает важным свойством - обеспечивает наследование знаний. Существующие системы чаще используют два других подхода. В соответствии с первым подходом каждая новая система проектируется «с чистого листа», т.е. не применяется каких либо специальных мер по накоплению знаний. Второй подход предполагает использования знаний о типовых архитектурных решениях.

К рассмотренному процессу проектирования предъявляется ряд требований. В частности, процесс должен позволять проектировать семейства и линейки систем, варьировать их функциональность. Также необходимо предусмотреть возможность выполнения нескольких итераций проектирования. Перечисленным требованиям удовлетворяют методологии, которые можно отнести к группе гибких методологий проектирования, например, экстремальное программирование, Scrum, Lean Development, Crystal Methods и др. С точки зрения ориентации на работу с отдельными функциями предпочтительным представляется ориентация на Feature-Driven Development методологию [298].

### 4.3.2. Иерархия архитектурных фреймворков ППСМ

Архитектурные фреймворки (архитектурные каркасы) ППСМ организованы в виде иерархии (рисунок 4.5). На верхнем уровне располагается мегафреймворк (MMF), на следующем за ним уровне располагаются доменные метафреймворки (DF).

Доменные метафреймворки делятся на фреймворки, ориентированные на использование в определенных предметных областях, и фреймворки, ориентированные на решение определенных задач. Фреймворки, относящиеся к первой группе, можно определить как предметно-ориентированные метафреймворки (ПОМФ), а ко второй - как проблемно- (задачно-) ориентированные метафреймворки (ЗОМФ). ПОМФ ориентированы на решение задач конкретной предметной области и в явном виде не накладывают ограничений на используемые архитектурные решения. ЗОМФ, напротив, ориентированы на использование определенных платформ и, в явном виде не ограничивают области применения. Следует заметить, что как ПОМФ, так и ЗОМФ, являются метафреймворками и предназначены не для генерации конкретных архитектур, а для генерации частных фреймворков (ЧФ).

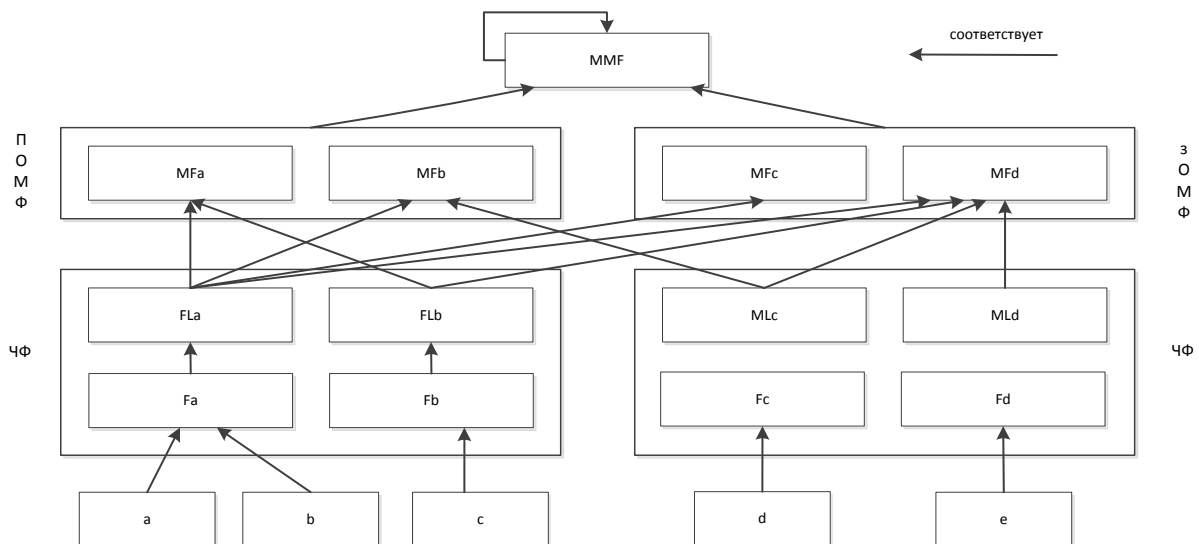


Рисунок 4.5 – Иерархия фреймворков ППС

Можно выделить два типа ЧФ: фреймворки, предназначенные для порождения архитектур семейств и линеек систем (FL) и простые частные фреймворки (F). Использование метафреймворков (MF) не является обязательным. Если метафреймворки используются, то они могут использоваться как для порождения архитектур семейств и линеек систем, так и выступать в качестве метафреймворков для порождения частных фреймворков F-типа.

Каждая архитектура (a, b, c, d.) соответствует ровно одному фреймворку F. Каждый фреймворк типа FL или F может строиться на базе нескольких DF, наследуя разные архитектурные элементы (интересы, точки зрения, модели). Если фреймворк F-типа строится на



базе FL-фреймворка, то каждый фреймворк F-типа соответствует одному FL-фреймворку, а каждому FL-фреймворку может соответствовать произвольное число F- фреймворков.

Предлагаемая иерархия фреймворков позволяет строить модели ППСМ. Фреймворки обеспечивают формирование архитектурных моделей уровней M1 - M4, определяемых методологией MDE, и соответствующих им онтологических моделей. Построенные модели являются адаптированными, в них учтены требования предметных областей и решаемых в них задач.

В связи с изменчивостью условий синтеза моделей объектов мониторинга нет возможности построить один частный фреймворк, одну модель трансформации данных. Они строятся для заданных условий. При этом существующие решения могут использоваться для координации работ [285].

Фреймворк ППСМ может быть отнесен к классу интеграционных архитектурных фреймворков. Он наследуется от нашедших широкое применение фреймворков DoD [299], TOGAF [300], фреймворка Захмана [301], обобщенного фреймворка [302]. В рамках фреймворка ППСМ обеспечивается их интеграция и адаптация к задачам проектирования и сопровождения ППСМ.

В предлагаемых фреймворках при проектировании предусматривается использование знаний. Наибольший эффект от использования знаний достигается при проектировании различных семейств и линеек систем, которые реализуются на разных платформах и отличаются назначением.

#### **4.3.3. Архитектурные стили проектирования ППСМ**

Архитектурные стили, применяемые для построения конечных систем, определяются на уровне частных фреймворков ППСМ. Можно выделить следующие архитектурные стили, которые являются наиболее подходящими для построения ППСМ: объектные (компонентные) архитектуры (ООА); сервисно-ориентированные архитектуры (СОА), многоагентные архитектуры (МАО), комбинированные (смешанные) архитектуры.

Типовая ИС, имеющая ООА, строится на основе объектной (компонентной) шины. Возможности управления конфигурацией принципиально существуют, но они достаточно ограничены. Для объектной реализации фактически единственной альтернативой обеспечения гибкости в режиме выполнения является включение в состав системы кода, обеспечивающего поддержку реконфигурации.

СОА строится на базе сервисной шины, в качестве которой могут использоваться различные программные продукты, такие как очереди сообщений, брокеры сообщений;

корпоративные шины - Enterprise Service BUS (ESB), сервер процессов, фабрики бизнес-служб [303], исполнительные подсистемы семантического веб [304].

В рамках СОА подхода можно выделить статические и динамические архитектуры. Статические архитектуры работают с BPEL файлами, которые интерпретируются BPEL процессором. При использовании динамических архитектур БП формируются в динамике, для чего используется специальный процессор. Как статические, так и динамические архитектуры могут поддерживать работу с бизнес-правилами.

Статические СОА представляет собой совокупность сервисных модулей (S-модулей), которые реализуют как бизнес-сервисы так и интеллектуальные сервисы. Возможности статических СОА, не поддерживающих работу с правилами, аналогичны ООА.

Следующим по сложности решением являются СОА, поддерживающие работу с бизнес-правилами. Основная идея заключается во включении в текст BPEL файла специального фрагмента, который фактически представляет собой управляемый условный оператор, настраиваемый пользователем. Функционирование бизнес-сервисов описывается в терминах предметной области. Для интерпретации правил используется специальный процессор правил [159]. Можно выделить 2 альтернативных подхода к построению динамических СОА. В рамках «слабого» подхода используются две основные концепции: бизнес-сервисы и композитные бизнес-приложения. Композитные приложения строятся на основе бизнес-сервисов в динамике средствами специального процессора (динамического ассемблера), при этом используется модель бизнес-объектов и бизнес-политики. Формирование композитных приложений основано на использовании OWL онтологий. Ключевым моментом данного подхода следует считать ориентацию на стандартизацию доменных онтологий и стандартизацию состава и интерфейсов бизнес-сервисов. В качестве «сильного» подхода к реализации динамических СОА можно рассматривать семантические веб-сервисы (СВС). Идея СВС заключается в стандартизации структуры онтологий, используемых для описания веб-сервисов, которые иногда называют сервисными онтологиями. В качестве стандарта на сегодняшний день принят OWL-S [305].

Одной из быстро развивающихся информационных технологий является технология построения мультиагентных систем (МАС) [306]. Типовая многоагентная система представляет собой множество агентов. В зависимости от уровня интеллекта можно выделить следующие основные классы агентов: реактивные агенты, агенты, управляемые целями, агенты, архитектуры которых основаны на полезности, обучающиеся агенты. С точки зрения реализации интеллектуальных возможностей в системах многоагентные системы с реактивными агентами не имеют каких-либо видимых преимуществ по сравнению с СОА системами. Агенты, управляемые целями, реализуют более сложное поведение. Их функционирование описывается в терминах целей, которые должны быть достигнуты. При этом агент выполняет такие действия как

определение желаемого конечного состояния, поиск путей достижения желаемого состояния (формирование плана) и реализация плана. При использовании агентов, управляемых целями, можно строить достаточно зрелые системы, обладающие высоким и очень высоким уровнем интеллекта.

Однако, несмотря на очень значительные успехи, достигнутые в области теории и практики построения МАС, по уровню инструментальной поддержки они существенно отстают от СОА. Таким образом, сильной стороной МАС является возможность реализовывать сложные модели поведения, а сильной стороной СОА является наличие возможности работы с БП, которые могут формироваться как в статическом режиме, так и в динамике [307]. В связи с этим представляется целесообразным рассматривать эти технологии как взаимодополняющие [308]. Можно выделить 2 альтернативных подхода к построению комбинированных архитектур: агентно-сервисные архитектуры, сервисно-агентные архитектуры.

В первом случае на верхнем уровне работают агенты, а на нижнем располагаются веб-сервисы. Во втором случае на верхнем уровне работают веб-сервисы, а на агентов возлагается решение отдельных задач. Основная идея агентно-сервисного подхода состоит в использовании на верхнем уровне агентов, обладающих более высокими «интеллектуальными способностями» для управления БП. СОА используются на нижнем уровне, и крайне желательно, чтобы они были стандартизированы в рамках отрасли или более узких рамках. Сервисно-агентный подход предполагает использование агентных технологий при построении отдельных подсистем. Следует отметить, что технические проблемы, связанные с сопряжением веб-сервисов и агентов, отсутствуют. Эта задача решается посредством использования агентов-шлюзов, к которым, с одной стороны, можно обращаться как к веб-сервисам, а, с другой стороны, через эти шлюзы можно посылать сообщения агентам.

При проектировании ППСМ предлагается перейти от 2-х слойной к 3-х слойной архитектуре, которая основана на применении комбинированного сервисного и агентного подходов и включает следующие слои (уровни): уровень элементарных веб-сервисов и бизнес-сервисов, уровень агентного управления, уровень БП верхнего уровня.

Трехслойная сервисно-агентно-сервисная архитектура строится в соответствии со следующими основными принципами.

1. Основные усилия в области стандартизации и унификации направляются на стандартизацию веб-сервисов и бизнес-сервисов, работающих на нижнем уровне.
2. Разделение БП на БП нижнего и верхнего уровня позволяют упростить разработку БП за счет разделения их на относительно стабильную и переменные части.
3. Стандартные механизмы и инструменты работы с бизнес-правилами можно использовать как на нижнем, так и на верхнем уровне.

4. Агенты работают на промежуточном уровне. Настройка поведения агентов может осуществляться либо с помощью встроенной системы работы с правилами, либо посредством редактирования онтологии, хранящей описания поведений. Могут применяться как реактивные, так и интеллектуальные агенты в зависимости от используемых алгоритмов управления.

#### 4.3.4. Сервисно-агентно-сервисная архитектура ППСМ

Для ППСМ в качестве базовой архитектуры предложена архитектура, приведенная на рисунке 4.6. Архитектура предусматривает наличие 2-х сегментов, соединенных с помощью моста. Кроме того, у ППСМ имеется централизованное хранилище данных и метаданных – реестр и репозитарий (R&R). Первый сегмент формируется на базе некоторой среды интеграции, в качестве которой может выступать, например, корпоративная сервисная шина или сервер интеграции [307].

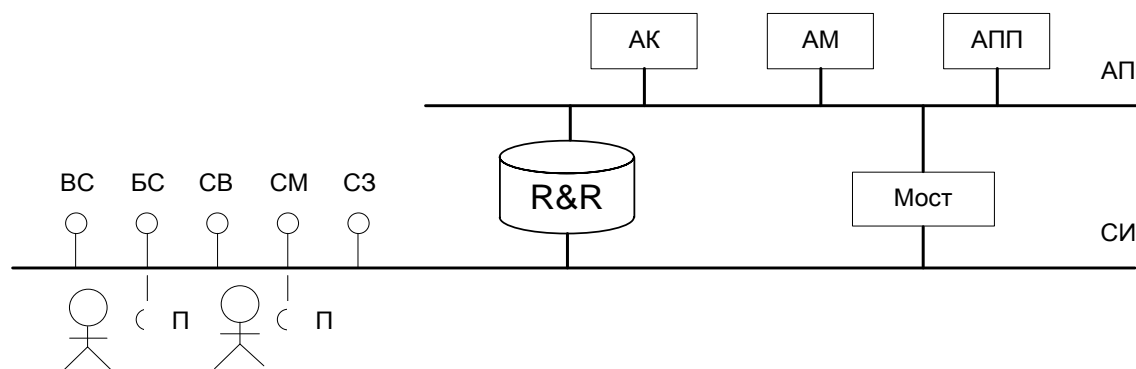


Рисунок 4.6 – Типовая архитектура ППСМ

На первом (нижнем) уровне работают элементарные веб-сервисы (ВС), реализующие собственно обработку, которые компонуются посредством WPEL и могут быть доступными как бизнес-сервисы (БС). На нижнем уровне также располагаются сервисы визуализации (СВ), сервисы мониторинга (СМ), а также сервисы работы со знаниями (СЗ). В качестве ВС могут выступать сервисы предварительной обработки, сервисы сегментации, сервисы кластеризации, сервисы классификации, сервисы поддержки принятия решений. Примерами БС могут служить сервисы обработки данных. СВ обеспечивают доступ к средствам визуализации в качестве которых могут использоваться порталы. СМ ориентированы на поддержку мониторинга состояния процессов обработки и анализа данных.

На втором уровне работают агенты, управляющие процессом обработки данных преимущественно посредством вызова бизнес-функций, хотя, в случае необходимости, агенты могут получать доступ к веб-сервисам. На уровне реализуются следующие типы агентов: агенты-координаторы (АК), агенты, выполняющие функции персональных помощников (АПП), агенты, отвечающие за мониторинг БП (АМ). Агентам ставится в соответствие набор поведений. Управление поведением реактивных агентов осуществляется с помощью правил, а

интеллектуальных - за счет редактирования онтологии поведений. Агенты, принадлежащие агентному слою, реализуют свое поведение посредством обмена сообщениями и вызова бизнес-функций через шлюзы.

Бизнес-логика инкапсулируется в веб-сервисах. Для работы агентов веб-сервисы должны быть стандартизированы. Веб-сервисы объединяются в стандартные бизнес-сервисы.

На третьем (верхнем) уровне работают БП верхнего уровня. На верхнем уровне для оркестровки БП целесообразно использовать статические БП (BPEL). На нижнем уровне можно использовать как статические, так и динамические механизмы формирования БП. Возможно использование динамического ассемблера. Использование семантических веб-сервисов на основе, например, OWL-S, ориентированных, прежде всего, на работу в открытых мирах, в общем случае, не является целесообразным. Однако при наличии доступной инструментальной поддержки использование семантических методов композиции веб-сервисов может оказаться достаточно эффективным. Основным доводом в пользу использования динамических методов компоновки веб-сервисов является возможность оперативно добавлять или удалять веб-сервисы. При большом количестве веб-сервисов отслеживать их использование достаточно сложно.

ППСМ (рисунок 4.7) представляют собой локальную сеть (ЛС), к которой подключены рабочие станции (РС). Рабочие станции являются рабочими местами пользователей ППСМ. Пользователями могут являться: эксперты, отвечающие за решение прикладных задач (Э), координаторы (К), руководители, определяющие состав решаемых задач и контролирующие их выполнение (БА), системные администраторы (СА).

Предусмотрено несколько типовых вариантов работы системы. При решении крупномасштабных задач в оперативном режиме получаемые данные в БД обрабатываются параллельно всеми РС. Если мощности РС недостаточно, то можно использовать выделенные серверы обработки и серверы БД (С). Эксперты могут наблюдать за ходом обработки данных, поступающих от различных источников. Координатор может наблюдать за ходом решения задач.

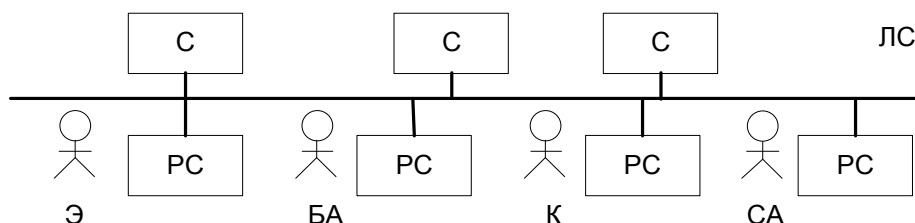


Рисунок 4.7 – Типовая структура ППСМ

Исходные данные находятся в БД, поэтому задачи или отдельные подзадачи могут решаться многократно, например, с целью отладки алгоритмов и БП.

При решении крупномасштабных задач в отложенном режиме система может работать следующим образом. Координатор запускает глобальный БП. БП верхнего уровня через шлюзы

посылает сообщения агентам о начале работы. При решении крупномасштабных задач запускаются все агенты, при решении локальных – один или несколько агентов. Агент отвечает за решение одной или нескольких подзадач. Для сложных систем агент может создавать агентов более низкого уровня. Агенты могут координировать свои действия.

Предлагаемая структура дает возможность оперативно изменять состав и функциональность веб-сервисов нижнего уровня, возможность реализовывать разумное поведение за счет использования интеллектуальных агентов.

Создаваемые системы легко «расширяются» и «сжимаются». При максимальной конфигурации системы могут быть использованы при работе с очень сложными объектами. «Сжатие» возможно, например, за счет полного или частичного исключения агентного слоя и (или) отказа от использования бизнес-сервисов. Можно также отказаться от использования динамических механизмов генерации БП нижнего уровня.

Следует отметить, что системы позволяют наращивать функциональность и «уровень интеллекта» практически без внесения изменений в программный код за счет расширения / уточнения описаний БП, настройки бизнес-правил и редактирования онтологий.

Стандартизация систем поддерживается на уровне веб-сервисов низкого уровня, на уровне бизнес-сервисов и на уровне библиотек поведений агентов.

#### **4.4. Архитектурные метрики систем ППСМ**

В основу построения системы архитектурных метрик ППСМ положены следующие принципы.

1. Метрики являются атрибутами архитектурной модели.
2. Каждая из архитектурных моделей может иметь произвольное число метрик в диапазоне от 1 до N.
3. Метрика должна быть востребована, по крайней мере, одной заинтересованной стороной.
4. Метрики снимаются по запросу одной или нескольких заинтересованных сторон.
5. Мера востребованности метрики является метаметрикой.
6. В качестве потребителей (заинтересованных сторон) метрик могут выступать как люди, так и технические системы.

Метрики должны удовлетворять следующим основным требованиям.

1. Система метрик должна быть функционально полной, но не избыточной.
2. Метрики должны быть представлены в форме, воспринимаемой заинтересованной стороной, т.е. в терминах модели, которой принадлежит данная метрика.

3. Стоимость метрики должна удовлетворять системным ограничениям. Стоимость метрики может выражаться разными способами, в частности, в денежном исчислении, в терминах уменьшения производительности контролируемой системы и т. п.

Возможная схема классификации метрик, образующая систему метрик, ориентированных на использование применительно к ППСМ, приведена на рисунке 4.8. Снимаемые метрики можно классифицировать по следующим основным признакам: аспекты, уровень общности задачи, сфера применения, назначение метрик.

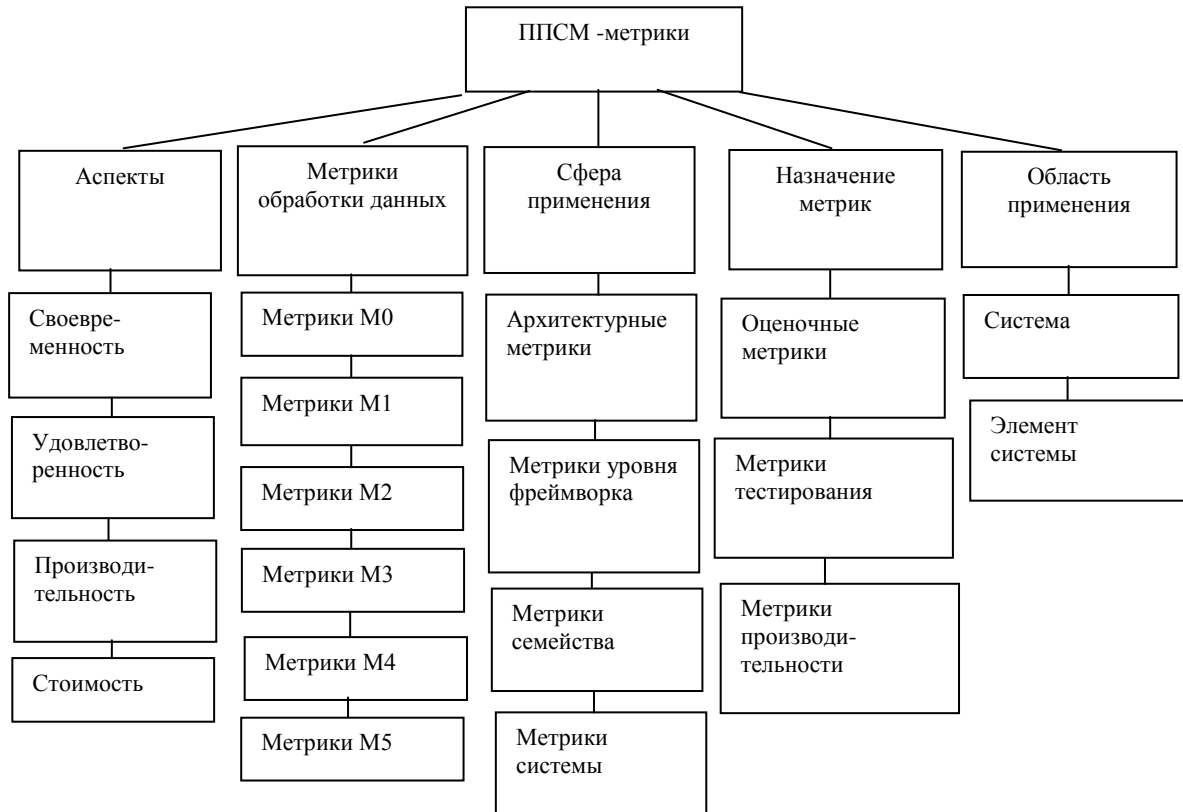


Рисунок 4.8 – Архитектурные метрики ППСМ

Можно выделить 4 основные группы параметров, которые соответствуют четырем основным группам требований, специфичным для ППСМ. К ним относятся: своевременность (Timeless) получения данных, удовлетворенность (Confidence) заинтересованной стороны (сторон) полученными данными, производительность (Performance), пропускная способность (Throughput) системы (подсистемы), стоимость (Cost).

С точки зрения сферы применения, метрики можно разделить на архитектурные метрики, метрики уровня фреймворка, метрики, относящиеся к семейству архитектур и метрики, относящиеся к конкретной системе. Уровень общности определяется уровнем, на котором выполняется обработка данных.

С точки зрения назначения, т.е. с точки зрения того, кто является потребителем метрик, можно выделить следующие основные группы метрик: оценочные метрики (Evaluation metrics), метрики тестирования (Testing metrics) и метрики производительности (Performance metrics).

Оценочные метрики – это, прежде всего, метрики разработчика, метрики тестирования – это метрики, которые позволяют сравнивать различные системы, т.е. это метрики системного архитектора, а метрики производительности – это метрики конечного пользователя, которые показывают в какой степени система может повысить эффективность процесса решения прикладных задач конечным пользователем [309].

Для каждого из уровней обработки существует собственная система метрик. В общем виде система метрик обработки для ППСМ показана на рисунке 4.9. На рисунке 4.9 L0 – L5 – модули, реализующие обработку на различных уровнях, а m0, m1, m2, m3 – метрики для соответствующих уровней. M0 – M3 - модули обработки метрик уровней L0 – L3. Обработка и оценка метрик, полученных с использованием модулей M0 – M3, выполняется в рамках модуля L4. Применительно к процессу обработки метрик обычно определяют пару значений: желаемое значение и минимально достаточное.

На такие системы многоуровневой обработки можно смотреть снизу вверх и сверху вниз. Движение сверху вниз – это процесс удовлетворения потребностей заинтересованных сторон, а движение снизу вверх – процесс устранения неопределенности [310].

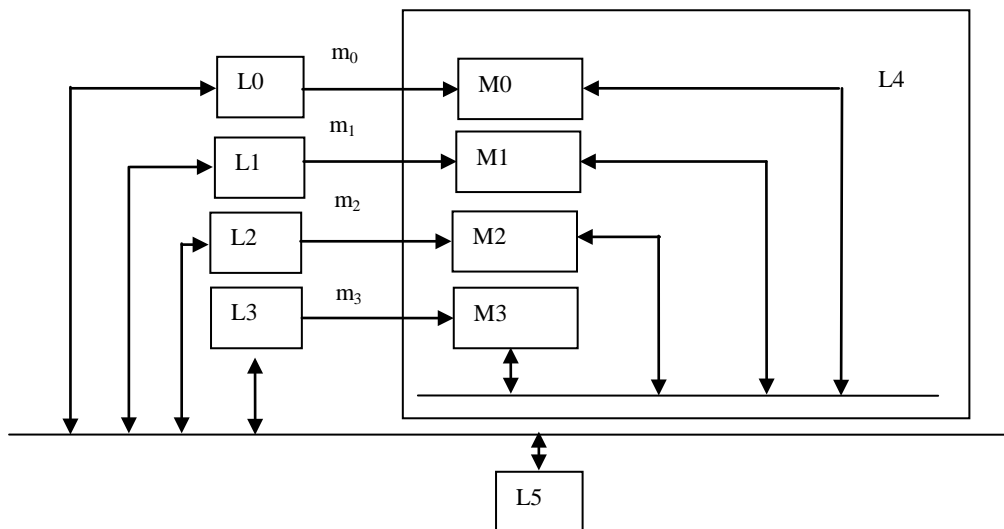


Рисунок 4.9 – Метрики обработки для ППСМ

Можно определить 3 основные цели снятия метрик обработки данных:

- получение оценочных метрик в интересах разработчиков системы;
- получение метрик, снимаемых в процессе тестирования, например, для получения сравнительных характеристик различных систем;



- получение метрик производительности, которые снимаются в интересах конечного пользователя.

Метрики могут относиться к классу систем в целом, к отдельным системам, к отдельным подсистемам или отдельным уровням обработки.

К типовым метрикам для отдельных уровней относятся следующие.

Метрики Уровня 0 (M0) – это метрики, связанные с точностью измерений. Это могут быть, например, ошибки измерения расстояния до объекта, ошибки измерения географических координат.

Метрики Уровня 1 (M1) – это метрики, связанные с такими понятиями как обнаружение, опознавание, классификация, идентификация и т. п. Для систем, решающих задачи, связанные с обнаружением и идентификацией объектов, в качестве типовых метрик могут выступать такие метрики как:

- вероятность обнаружения (Probability of Detection (PD)) – отношение числа обнаруженных объектов (ND) к общему числу объектов (NDO);

- вероятность распознавания (Probability of recognition (PR)) – отношение числа успешных распознаваний (NR) к общему числу попыток распознавания (NRO) ( $PR = NR / NRO$ );

- вероятность правильной идентификации (Probability of Correct Identification (PID)) — отношение числа правильных распознаваний (NID) к общему числу попыток распознавания (NIDO) ( $PID = NID / NIDO$ ).

Если речь идет, например, об отслеживании некоторой траектории (Track) движущегося объекта, то могут использоваться такие метрики как:

- вероятность обнаружения трека (Probability of Track Detection (PDT)) – отношение найденных траекторий (NDT) к общему числу реально существующих траекторий (NTT) ( $PDT = NDT / NTT$ );

- доля неправильно определенных траекторий (Track False Alarm Fraction (FT)) - отношение правильно определенных траекторий (NDT) к общему числу анализируемых траекторий (NTT) ( $FT = NFT / NTT$ ).

Метрики Уровня 2 (M2) базируются на потребностях в данных и результатах их обработки, которые запрашивают заинтересованные стороны. Запросы могут касаться текущей ситуации, ситуации, относящейся к определенным моментам в прошлом, запросы на общие оценки ситуаций или их конкретных аспектов. Оценивание процесса формирования данных о ситуации и оценка ситуации выполняется исходя из запроса пользователя относительно той или иной ситуации.

Конечный пользователь предъявляет к данным следующие основные требования: предоставляемые данные должны быть полными; данные должны представляться в форме, допускающей их эффективную обработку (восприятие); данные должны быть полезными -

лишние данные должны быть отфильтрованы; данные должны предоставляться своевременно; данные должны быть точными; данные должны быть достоверными; данные должны быть структурированными.

Метрики можно разделить на 2 группы: метрики нижнего уровня и метрики верхнего уровня. Метрики нижнего уровня получаются в результате агрегации метрик ниже лежащих уровней, а метрики верхнего уровня – это данные, которые извлекаются из результатов снятия метрик нижнего уровня. Например, на основе метрик точности для группы объектов формулируются метрики точности для группового объекта. На основе информации о месте нахождения группы объектов с учетом ошибок отдельных операторов и с учетом погрешности измерений можно получить информацию о точности определения места нахождения такой группы.

Основная идея метрик уровня M2 – это оценка удовлетворенности заинтересованных сторон качеством и стоимостью информации, которую они хотят получать.

Метрики Уровня 3 (M3). Метрики третьего уровня ориентированы на оценку угроз. Традиционно (начиная с JDL-моделей слияния данных) такая оценка выполняется в следующих терминах: выигрыши (Benefits), стоимость (Costs), риски (Risks). Поскольку процесс обработки данных обычно направлен либо на уменьшение неопределенности, либо на увеличение получаемых данных, метрики уровня 3 должны давать оценки качества принимаемых решений в терминах рисков [98].

Типовыми метриками уровня M3 можно считать:

- степень готовности заинтересованной стороны к риску;
- уязвимость - отношение числа неблагоприятных вариантов развития событий к общему числу вариантов развития событий, с учетом вероятностей и весов;
- живучесть – отношение числа угроз, которые не являются опасными для системы к общему числу угроз с учетом вероятностей и опасности угроз;
- отношение числа благоприятных вариантов развития событий к общему числу вариантов развития событий с учетом вероятностей и весов.

Одним из важнейших компонент метрик уровня 3 является показатель устаревания во времени (Time decay), в качестве которого выступает некоторая экспоненциальная зависимость, описывающая процесс устаревания данных во времени и (или) при условии появления информации об определенных событиях. Увеличение задержек при получении информации, в общем случае, приводит к большей неопределенности и большим рискам.

В качестве метрики качества функционирования системы может выступать процент реализованных планов (Percentage of plans achieved (PPA)) - отношение числа реализованных планов к общему числу планов ( $PPA = NPC / NPCR$ ).

К третьему уровню также относятся следующие метрики:

- отношение Выгода/Затраты (Benefit to Cost Ratio Confidence (BC)) ( $BC = B / C$ ), метрики B и C выражаются в терминах предметных областей;
- выживаемость (Survivability), которая определяется как показатель степени готовности заинтересованной стороны к риску.

Метрики Уровня 4 (M4). Уровень 4 занимает особое место, поскольку представляет собой встроенную модель, которая реализует практически полную функциональность модели трансформаций. При этом в качестве внешней среды выступает сама обслуживаемая система. Поэтому функционирование подсистемы управления процессами может быть оценено в тех же терминах, что и функционирование любой ППСМ, т.е. в терминах своевременность (Timeliness), точность (Accuracy), пропускная способность (Throughput), стоимость (Cost), уровень доверия (Confidence).

Метрики уровня M5 (M5) – это метрики, которые должны оценивать эффективность функционирования в терминах человеко-машинной системы. В рамках такого подхода оценка эффективности самой ППСМ с точки зрения пользователя выполняется в терминах удовлетворенности пользователя системой, т.е. в терминах того, что система предоставляет пользователю конкретные данные, необходимые для решения прикладных задач в адекватной форме, принятия решений, а также средства для реализации принятых решений. В частности, в [98] для оценки степени удовлетворенности пользователя предлагается использовать совокупность взвешенных показателей, таких как предсказуемость, надежность предоставляемых данных, степень доверия.

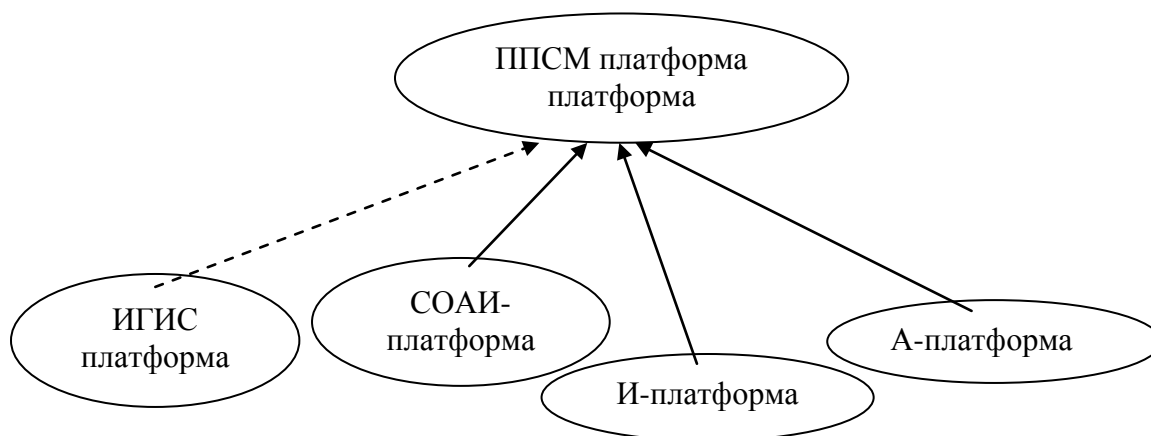
#### 4.5. Платформа ППСМ

Платформа ППСМ – это согласованная совокупность технологических, программных и инструментальных средств, которые могут использоваться при решении прикладных задач доменного уровня. В состав ППСМ платформы могут входить библиотеки алгоритмов и методов обработки данных, онтологии, описывающие технические и природные объекты, шаблоны разного уровня, другие платформы, которые могут оказаться востребованными при разработке ППСМ.

В широком смысле ППСМ платформу можно рассматривать как одну из возможных реализаций ППСМ.

При взгляде на платформу с позиций ИТ (программная точка зрения), ее можно соотнести с понятием фабрики в смысле [303]. Такое соответствие определяется схожестью функций, которые реализует платформа и фабрика. Платформа как фабрика представляет собой мета систему, используя которую можно порождать произвольное множество целевых ППСМ по их описаниям. Различные точки зрения можно рассматривать как архитектурные перспективы в смысле [311].

Еще одной точкой зрения является интеграционная точка зрения. Такая точка зрения рассматривает возможности платформы по интеграции доменных моделей, готовых технологий, сторонних платформ. На более низком уровне может рассматриваться интеграция онтологий, программных компонент, отдельных сервисов и т.д. На рисунке 4.10 показана возможная иерархия наследования платформ, позволяющая построить ППСМ, обладающую возможностями адаптивной интеллектуальной обработки данных на основе ГИС технологий.



Рисунке 4.10 – Иерархия наследования платформы ППСМ

На рисунке 4.10 используются следующие обозначения: ИГИС платформа - интеллектуальная геоинформационная платформа, СОАИ платформа - платформа, построенная на базе прикладных систем обработки и анализа результатов измерений, А-платформа - платформа для решения задач адаптации, И-платформ - интеллектуальная платформа. В настоящее время реально существуют и активно развиваются 2 платформы: ИГИС платформа [312], [313] и СОАИ платформа [191]. Готовые А-платформы в настоящее время отсутствуют, однако разработано большое количество отдельных моделей, методов и инструментов, ориентированных на реализацию механизмов адаптации. Существует значительное число И-платформ. В данном случае под И-платформой понимается интеграционная платформа, включающая множество отдельных платформ. К таким платформам относятся платформы машинного обучения, интеллектуального анализа данных, семантические платформы, экспертные системы и другие.

Использование ППСМ платформы позволяет:

а) разрабатывать и реализовывать системы построения моделей объектов, обеспечивающие решение прикладных задач. Для этого платформа предоставляет средства для оперативного сбора актуальных данных от различных поставщиков, инструменты искусственного интеллекта, средства имитационного и математического моделирования;

б) создавать сложные системы, работающие с несколькими предметными областями. Работа с несколькими предметными областями осуществляется за счет интеграции онтологий,

описывающих предметные области. При интеграции используются текстовые и графические редакторы онтологий;

в) создавать простые для использования системы, ориентированные на потребителей с различной квалификацией. Наличие графических интерфейсов, например, ГИС интерфейсов, позволяет создавать системы, ориентированные на пользователей, не являющихся специалистами в обработке и анализе данных и в области IT в целом;

г) выполнять быструю настройку систем при изменении требований к ним или необходимости портирования таких систем в другую предметную область. Совокупность визуальных сред позволяет вносить изменения в логику работы систем без внесения изменений в программный код. Примерами таких сред являются редакторы процессов;

д) накапливать и эффективно организовывать работу со знаниями при разработке систем. Знания о ППСМ формируемые и используемые на этапах проектирования и создания размещаются в онтологиях. Описание таких онтологий и рекомендации по их применению размещаются в платформе;

е) использовать в ППСМ специализированные технологии. В качестве таких технологий могут выступать технологии работы с большим объемом данных, технологии облачных вычислений и другие готовые технологии, которые могут быть интегрированы в ППСМ с учетом выбранных архитектурных решений и поддерживаемого технологического стека.

Применение платформы ППСМ позволяет увеличить надежность и скорость разработки конечных систем, в разы сократить затраты ресурсов за счет использования предоставляемых платформой готовых решений. В их состав входят решения, обеспечивающие построение моделей объектов мониторинга, а также решения, типовые для информационных систем, например, [314]–[316]. Готовые компоненты и элементы содержат незначительное число программных ошибок, а готовые среды требуют минимальной интеграционной отладки и т.д.

#### **4.5.1. Сервисно-процессное представление платформы ППСМ**

Платформа ППСМ и формируемые на ее основе системы позволяют строить гибкие процессы при построении моделей объектов по данным мониторинга.

Расширенный состав понятий, относящихся к построению и реализации гибких процессов, а также взаимосвязи между ними приведены на рисунке 4.11. К основным понятиями, связанными с гибкими процессами, относятся технология, сервис, БП, бизнес-правило, скрипт, шаблон, политика, автомат, стратегия.

Технология – это неупорядоченный или частично упорядоченный набор шаблонов, методов, алгоритмов, паттернов, библиотек, сервисов, скриптов, а также набор правил и ограничений по их использованию. Технология может определяться для всего класса ППСМ,

отдельных его подклассов, может быть привязана к конкретной системе. На базе технологий строятся шаблоны. Шаблон – это описание последовательности действий по решению конкретной задачи. Шаблоны обычно представляют в виде графов, описывающих порядок выполнения действий в рамках конкретной технологии. Они могут строиться в статике или динамике. Шаблон можно рассматривать как абстрактный БП, используемый для генерации исполняемого БП. Таким образом, шаблон является частью технологии и описывает множество программных решений. По шаблону строится исполняемая модель, например, UML модель. Из исполняемой модели средствами существующих инструментальных средств [317] можно получить:

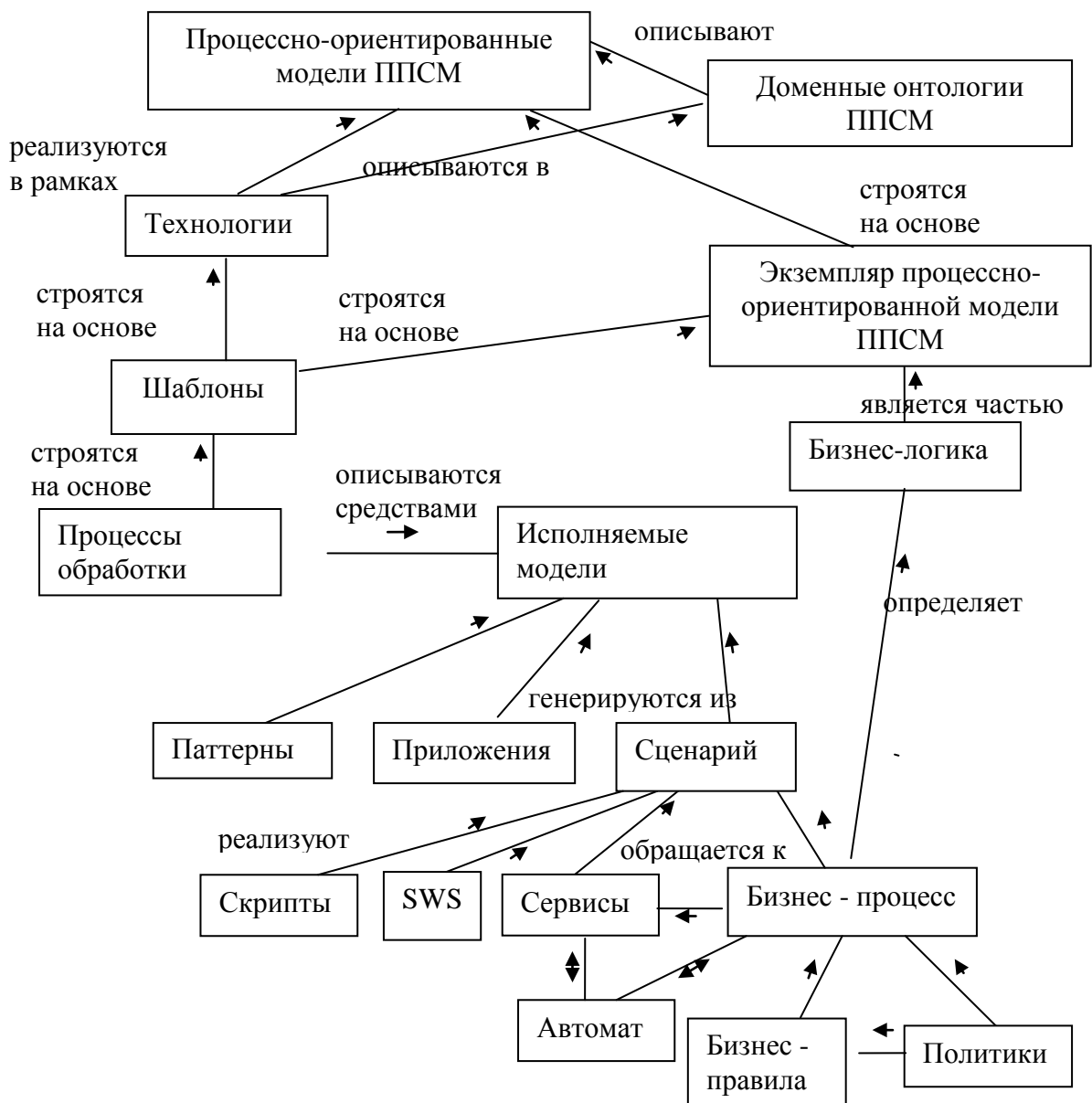


Рисунок 4.11 – Взаимосвязь понятий, определяющих гибкие процессы ППСМ

- программный код, который может быть оформлен как приложение, библиотечная функция, сервис;
- паттерн в стиле [318];

- сценарий, который может быть реализован как скрипт на том или ином языке описания сценариев, БП, набор бизнес-правил.

Под скриптом понимается описание сценария на некотором интерпретируемом языке. Скрипт может написать не только разработчик, но и администратор и даже пользователь. Одним из основных вариантов описания сценариев являются БП. Далее именно этому варианту уделяется основное внимание, в частности, вводятся понятия сервисов, правил и другие. Отметим, что большинство из этих понятий также применимы к скриптам и сценариям.

БП – это некоторая упорядоченная последовательность обращения к сервисам. Извне БП выглядит как сервис. Термин сервис понимается в самом широком смысле и определяется как некоторая адресуемая точка в информационном пространстве, к которой можно обращаться целью выполнения некоторых действий. Последовательность обращений может определяться бизнес-правилами. Бизнес-правило – правило типа “если - то”, сформулированное в терминах предметной области. Политика – упорядоченное множество БП или бизнес-правил, которые следует применять (запускать) для достижения определенной цели в определенных условиях. Правила обычно имеют следующий формат: для достижения цели X при условии Y следует запустить БП Z.

Автомат – это некоторая абстрактная или физическая сущность, функционирование которой описывается в терминах входных сигналов, выходных сигналов и внутреннего состояния. В виде автоматов могут представляться отдельные элементы БП и, наоборот, БП может являться реализацией автомата.

При процессно-сервисном представлении ППСМ описывается как совокупность сервисов и БП разного уровня. Сервисы самого низкого уровня носят название элементарных сервисов, а сервисы более высокого – композитных сервисов. При вызове сервиса элементарные и композитные сервисы неразличимы.

Процессно-сервисное представление позволяет формировать функциональные требования к целевым системам на этапе их проектирования, описывать механизмы адаптации, логику функционирования целевых систем при использовании СОА решений и других.

Для ППСМ процессно-сервисное представление может считаться основным представлением. Это обусловлено следующими причинами:

- большинство целевых ППСМ реализуются как СОА системы;
- в рамках СОА систем лучше всего реализуются механизмы адаптации.

К основным группам сервисов, используемых при построении и реализации гибких процессов ППСМ, относятся функциональные сервисы, сервисы управления БП, сервисы доступа к данным, вспомогательные сервисы (рисунок 4.12). Отметим, что совокупность этих сервисов определяет типовую сервисно-ориентированную архитектуру ППСМ.

Функциональные сервисы отвечают за реализацию требуемой функциональности систем. Их можно разделить на инфраструктурные, системные и бизнес-сервисы. Инфраструктурные сервисы – это сервисы низкого уровня, которые, как правило, реализуются как программно-аппаратные. Они обычно относятся к определенному уровню модели трансформаций, но отдельный инфраструктурный сервис, как правило, не закрывает какой-то уровень в целом. Системный сервис – это обычно композитный сервис, который реализует конкретный уровень. Их интерфейсы формируются в терминах информационной модели трансформаций. Бизнес-сервисы также являются композитными. Бизнес-сервис определяется в бизнес терминах, т.е. терминах прикладных областей. Обычно, бизнес-сервисы покрывают несколько уровней модели.

Управляющие сервисы включают процессы генерации БП и сервисы выполнения (интерпретации), которые называют движками (Engine).

Сервисы доступа к данным реализуют доступ к источникам данных и знаний. Можно выделить следующие типы сервисов данных: сервисы извлечения/сохранения данных, сервисы извлечения знаний (Data Mining), сервисы анализа процессов (Process Mining). К этой группе относятся также сервисы доступа к историческим данным и внешним данным.



Рисунок 4.12 – Типовая сервисно-ориентированная архитектура систем ППСМ

Вспомогательные сервисы (сервисы адаптации) включают в себя сервисы мониторинга, оценки и реконфигурации.

Пример ППСМ, построенной по принципу СОА, показан на рисунке 4.13.



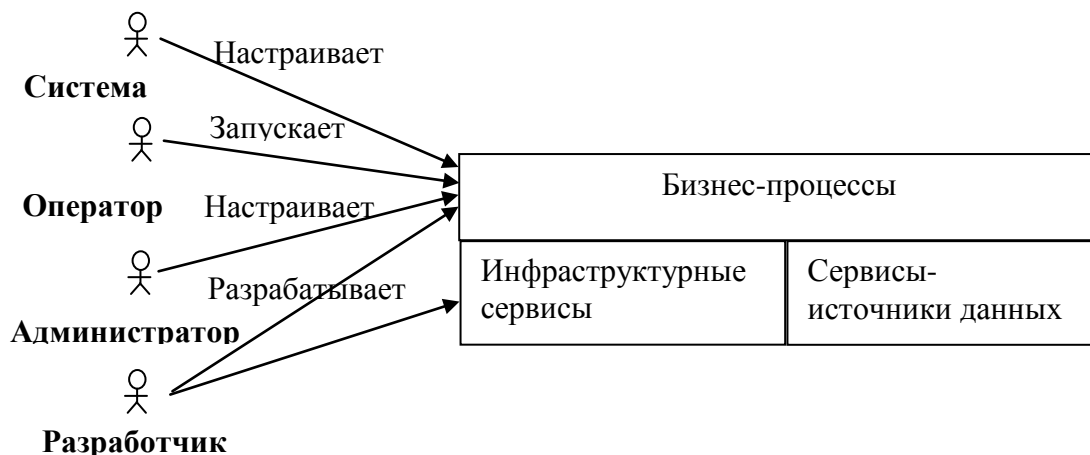


Рисунок 4.13 – Сервисная структура ППСМ

На нижнем уровне располагаются инфраструктурные сервисы и сервисы доступа к данным. Эти сервисы реализуются средствами стандартных ЯВУ и возможности их настройки обычно крайне ограничены. На верхнем уровне находятся БП, которые работают на 2х уровнях: системном уровне и бизнес-уровне. Настройки реализуются на уровне БП. В качестве заинтересованных сторон выступают: Разработчик, Администратор, Оператор (Пользователь) и Система. Разработчик отвечает за разработку как нижнего, так и верхнего уровней. Администратор имеет право настраивать БП. Оператор может только запускать БП. Система может настраивать БП.

Значительный интерес при реализации механизмов адаптации представляет вопрос управления БП. Такое управление может быть организовано разными способами. Классификация возможных подходов к реализации БП в ППСМ приведена на рисунке 4.14.

Возможные способы управления БП в ППСМ можно классифицировать по следующим признакам: используемая стратегия управления БП, используемое архитектурное решение и способ формирования БП.

Можно выделить 3 основных стратегии управления БП: потоковая, запросная и директивная стратегии. Потоковая стратегия предполагает, что БП развивается по мере поступления данных и обращение к сервисам осуществляется по мере готовности данных. Основное направление развития – от сырых данных к результату, который представляется в форме знания, например, знания о ситуации. При использовании запросной стратегии, напротив, инициатором запуска БП является заинтересованная сторона, которая обращается к системе с некоторым запросом, при этом не уточняется, какие именно сервисы будут задействованы для решения задачи. БП при этом реализуется в 2 прохода. На первом проходе определяется структура БП, а на втором – в потоковом режиме выполняется БП. Возможен подход, при котором БП строится по частям. Директивная стратегия предполагает, что последовательность обращения к сервисам определяет некоторый внешний монитор, который решает, когда будет запущен

конкретный сервис. При этом в явном виде не проверяется готовность данных и наличие запроса. Подобный монитор может работать, например, по таймеру. Следует заметить, что на практике чаще всего используется комбинация перечисленных выше стратегий.

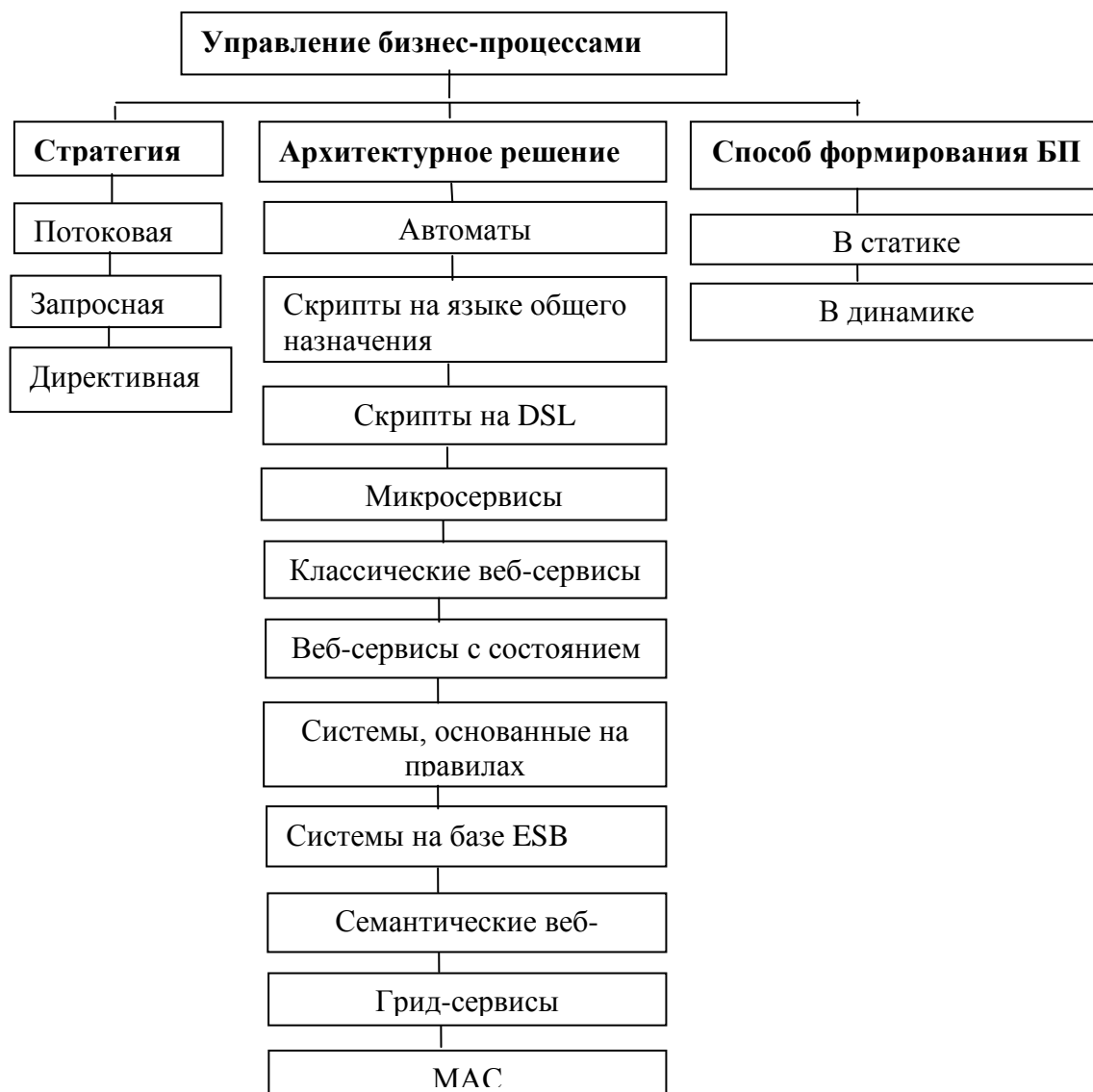


Рисунок 4.14 – Способы управления БП в ППСМ

В качестве типовых вариантов, которые могут применяться при построении конечных ППСМ, можно выделить следующие:

- использование централизованного монитора, построенного по принципу машины состояний (автомата);
- описание БП на некотором интерпретируемом ЯВУ;
- описание БП на доменно-ориентированном интерпретируемом языке (Domain Specific Language, DSL [294]);
- использование микросервисной архитектуры, при этом для управления БП могут использоваться разные подходы, но чаще всего используются автоматы;

- использование классических веб-сервисов, при этом для описания управления БП, как правило, используется язык BPEL;
- использование классических веб-сервисов с состояниями на базе WSRF [319];
- использование систем, основанных на правилах [320];
- использование систем на базе корпоративной сервисной шины (Enterprise Service Bus, ESB), при этом для управления БП, как правило, используется язык BPEL;
- семантические веб-сервисы, при этом для управления БП чаще всего используется подход на основе OWL-S [305];
- использование обычных грид-сервисов [319] или семантических грид-сервисов;
- использование мультиагентных систем (МАС).

БП могут формироваться как в статике, так и в динамике.

В таблице 4.6 приведены достоинства и недостатки отдельных подходов к управлению БП в ППСМ. В таблице 4.6 в графе “Область применения” используются следующие обозначения: 1) Конвейерные системы РВ 2) Системы отложенной обработки 3) Локальные ИУС 4) Распределенные ИУС 5) Информационно-управляющие системы.

#### 4.5.2. Другие представления ППСМ

Процессно-сервисное представление ППСМ может дополняться объектно-ориентированным (ООП), онтологическим (ОП), автоматным (АО) и другими представлениями. Допустимый состав представлений определяется архитектурными точками зрения [311]. Для примера рассмотрим некоторые из них.

Объектно-ориентированное представление, прежде всего, позволяет получать исполняемый код через механизмы MDA. Онтологическое представление обеспечивает реализацию интеллектуальной составляющей ППСМ, динамическое формирование БП, накопление архитектурного знания и другое. Автоматное представление используется при решении следующих задач:

- для описания без явной привязки к способу реализации;
- для описания механизмов функционирования ППСМ в формализованном виде;
- для доказательства корректности БП, в частности, БП, сгенерированных динамически;
- при использовании SOA архитектур обеспечивать возможность абстрагирования логики процессов, что на рисунке 4.11 отражается в виде двухсторонней связи между бизнес-процессом и автоматом.

Все эти представления могут являться как платформенно-зависимыми, так не иметь прямых связей с реализациями на конкретной технологической или программной платформе.

Таблица 4.6 – Достоинства и недостатки отдельных подходов к управлению БП в ППСМ

№	Архитектура	Достоинства	Недостатки	Область применения				
				1	2	3	4	5
1	Автоматы	Высокая скорость работы	Невысокая гибкость Слабая поддержка распределенных вычислений	+	+	+	-	x
	Скрипты на языке общего назначения	Относительная простота реализации	Сложность реализации механизмов адаптации Слабая поддержка распределенных вычислений	x	+	+	-	-
2	Скрипты на DSL	Не только администратор, но и пользователь может выполнять настройки	Для каждого предметного домена необходимо иметь собственный DSL Сложность поддержка распределенных вычислений	-	+	+	-	-
3	Микросервисы	Относительная простота реализации Способность эффективно работать с сетями датчиков Наличие готовых платформ	Невысокая скорость работы	x	+	+	x	+
4	Классические веб-сервисы	Высокая гибкость Наличие готовых платформ Хорошая поддержка распределенных вычислений	Невысокая скорость работы	-	+	-	+	x
5	Веб-сервисы с состоянием	Высокая гибкость Наличие готовых платформ Хорошая поддержка распределенных вычислений	Невысокая скорость работы Ограниченный набор инструментальных средств	-	x	x	+	x
6	Системы, основанные на правилах	Высокая гибкость	Слабая поддержка распределенных вычислений	-	x	+	x	-
7	Системы на базе ESB	Высокая гибкость Наличие готовых платформ Хорошая поддержка распределенных вычислений	Невысокая скорость работы	-	+	+	+	x

Продолжение таблицы 4.6

№	Архитектура	Достоинства	Недостатки	Область применения				
				1	2	3	4	5
8	Семантические веб-сервисы	Очень высокая гибкость	Сложность реализации Невысокая скорость работы Ограниченный набор инструментальных средств	-	x	x	+	+
9	Грид-сервисы	Возможность работы с большими данными	Сложность реализации механизмов работы с ситуациями Невысокая гибкость	-	+	x	+	-
19	МАС	Очень высокая гибкость Хорошая поддержка распределенных вычислений	Сложность реализации Невысокая скорость работы Ограниченный набор инструментальных средств	-	x	x	+	x

Переход между различными представлениями не является трудно разрешимой задачей. Трансформация между объектно-ориентированным и процессно-сервисным представлениями может быть реализована средствами UML. Для этого существуют UML диаграммы. То же самое можно сказать об автоматных моделях. Вопросы трансформации между онтологическим и объектно-ориентированными описаниями также подробно исследованы, в частности в [271]. Также в рамках UML имеются профили для работы с онтологиями [271]. Таким образом, можно утверждать, что дополнительные исследования, связанные с трансформацией представлений не требуются.

### 4.5.3. Механизмы и средства адаптации ППСМ

Общая идея реализации механизмов адаптации в системах ППСМ показана на рисунке 4.15.

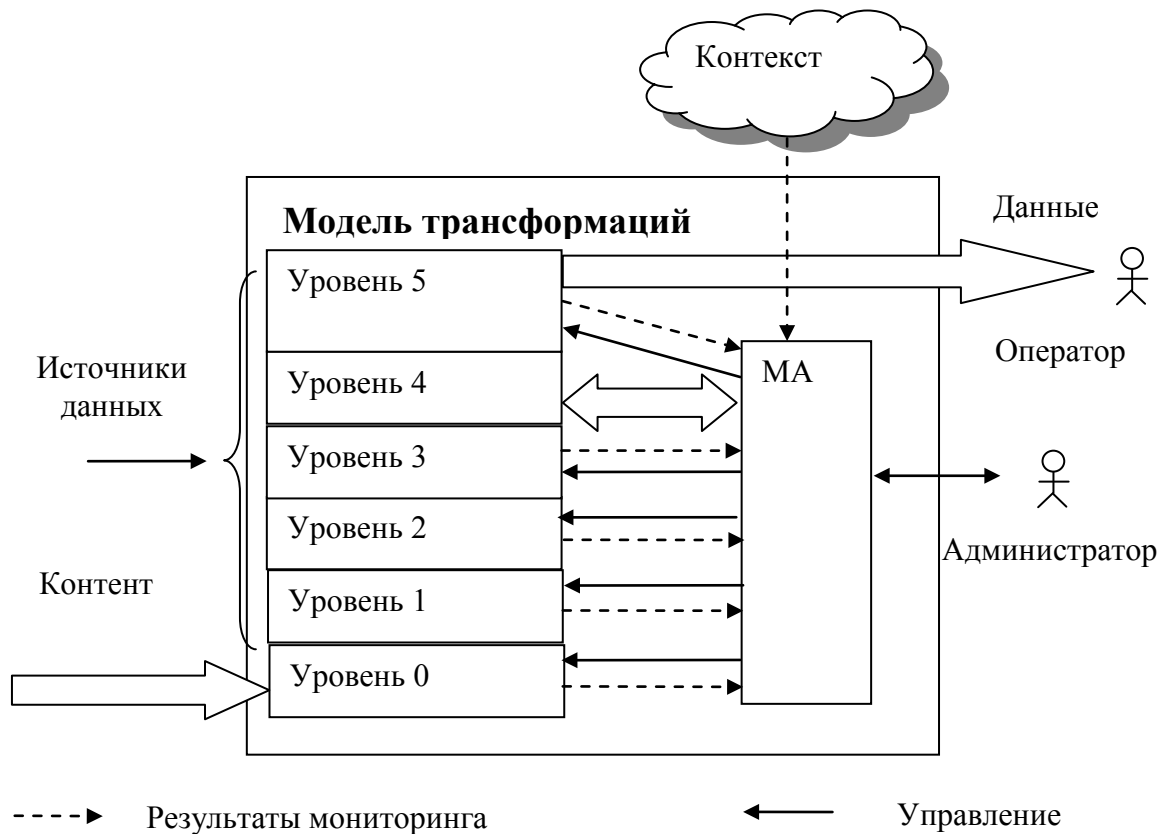


Рисунок 4.15 – Общая идея реализации механизмов адаптации в ППСМ

Механизмы адаптации в ППСМ необходимы для реализации всех уровней модели. Можно ожидать, что в значительной части адаптация будет обеспечиваться средствами сервисов 4 уровня– уровня поддержки принятия решений. Однако такие сервисы будут разнесены и по остальным уровням. В связи с этим целесообразным представляется организация сервисов, реализующих механизмы адаптации, в отдельный стек. Отметим, что вполне допустимы пересечения на уровне инфраструктурных сервисов.

В рамках ППСМ реализуются 4 типа адаптации: адаптация к специфике входного потока (контенту), адаптация к контексту, адаптация к требованиям и адаптация к ресурсам.

Для примера рассмотрим один из механизмов адаптации (МА), который может поддерживаться сервисами, относящимися к 4 уровню модели. Такой механизм может включать 4 фазы:

- 1) мониторинг состояния сервисов;
- 2) оценка текущего состояния системы;
- 3) принятие решения о необходимости адаптации;
- 4) выполнение, в случае необходимости, процедуры адаптации.

Способ получения информации о состоянии сервисов и выполнения настроек зависят от специфики используемой платформы.

Можно выделить 3 альтернативных подхода к реализации системы сервисов: централизованный, децентрализованный, иерархический, интегрированный с системой управления инфраструктурой.

При использовании централизованного подхода имеется одноуровневый БП, который внешне выглядит как один сервис, отвечающий за реализацию всех действий, связанных с адаптацией.

При использовании децентрализованного подхода имеется несколько независимых БП. В частности, может быть выделен отдельный БП, отвечающий за работу с контекстом. При использовании адаптивных алгоритмов процесс адаптации инкапсулируется в рамках одного (обычно 0 уровня).

При использовании иерархического подхода БП, отвечающий за адаптацию, имеет иерархическую структуру и может инкапсулировать подпроцессы, отвечающие, например, за отдельные уровни модели, что позволяет запускать эти подпроцессы параллельно. БП адаптации, отвечающий за адаптацию в рамках ППСМ, может быть интегрирован с БП, отвечающими за управление инфраструктурой (серверы, сетевая инфраструктура, виртуальные машины и т.п.).

Наличие механизма адаптации (рисунок 4.15) позволяет поддерживать различные схемы адаптации в ППСМ. Эти схемы строятся на основе возможных подходов к реализации адаптации в системах ППСМ (рисунок 4.16). Подходы можно классифицировать по следующим признакам: тип адаптации, аспект адаптации, механизмы, используемые для адаптации, используемое техническое решение, способ реализации и режим.

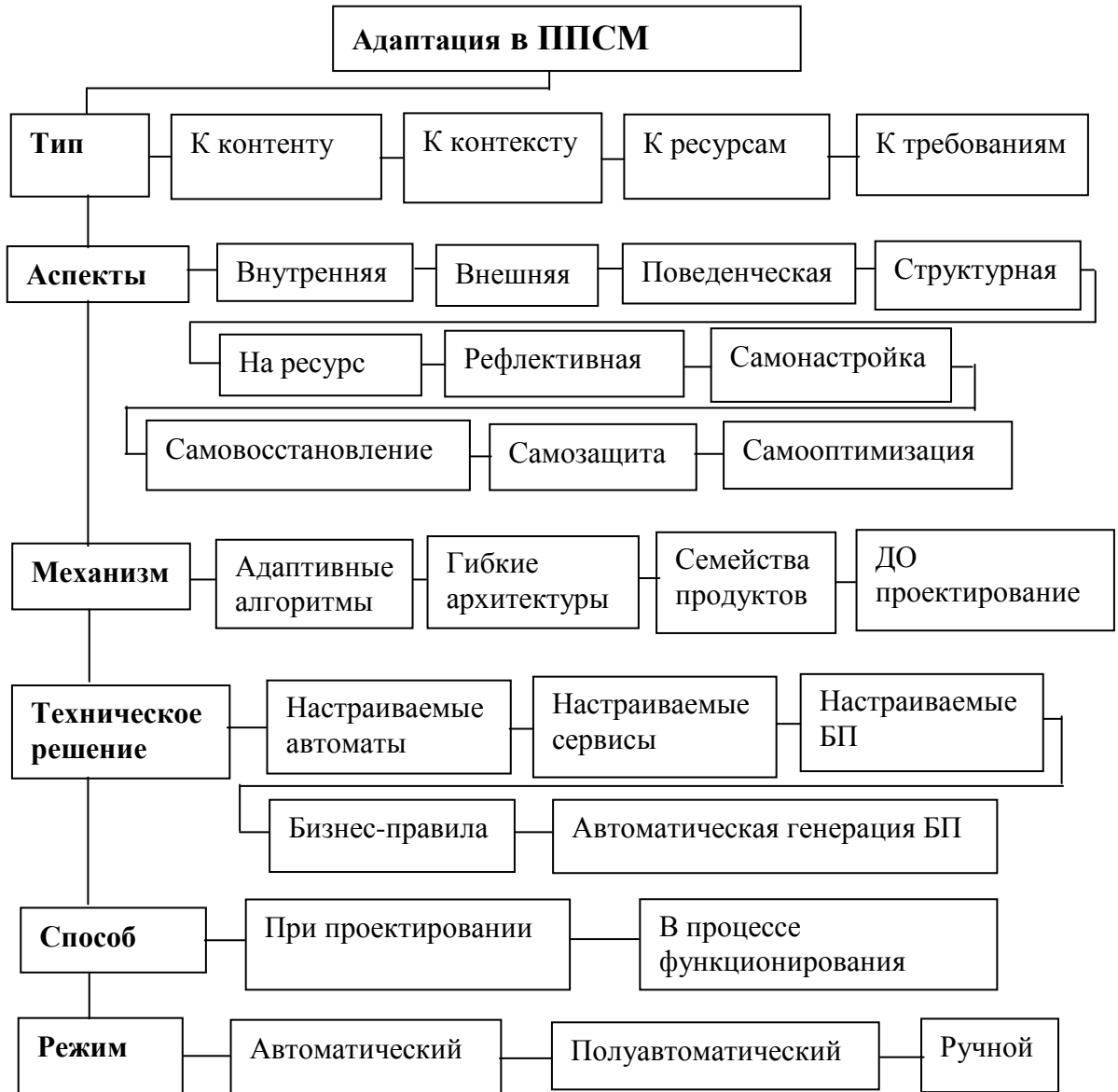


Рисунок 4.16. Подходы к реализации адаптации в ППСМ

Применительно к ППСМ используются 4 типа адаптации: адаптация к специфике входного сигнала (адаптивные алгоритмы), адаптация к контексту (внешним факторам), адаптация к доступным ресурсам, в первую очередь, инфраструктуре, и адаптация к требованиям. К возможным аспектам адаптации относятся: внутренняя адаптация, внешняя адаптация, поведенческая адаптация, структурная адаптация, способность настраиваться на имеющиеся ресурсы, рефлексивная адаптация, самонастройка, самовосстановление, самооптимизация. Следует заметить, что здесь имеются некоторые пересечения между отдельными аспектами. Это обосновано тем, что для выделенных аспектов имеются готовые механизмы. Для перечисленных аспектов сохранены авторские названия [254]. При изменении состава аспектов, потребуется их разработка, что не желательно.



Можно выделить 4 основных механизма адаптации: адаптивные алгоритмы, гибкие архитектуры, семейства и линейки систем и ДО проектирование.

Существует, по крайней мере, 4 типовых технических решения для реализации различных эффектов адаптации: настраиваемые автоматы, настраиваемые сервисы, настраиваемые БП, бизнес-правила и автоматическая генерация БП. Кроме того, возможны различные комбинации перечисленных подходов.

Идея настраиваемых автоматов состоит в том, что поведение автомата, описываемое в терминах внутренних состояний, переходов и выходов, может изменяться или его могут изменять извне. Такое техническое решение предполагает явную программную реализацию автомата.

Настройка сервисов возможна в случае, если сервисы обладают состоянием. В первую очередь, это относится к REST [321] сервисам и сервисам, поддерживающим WSRF [319]. Для ППСМ развивается подход, основанный на использовании настраиваемых БП. БП можно настроить, например, за счет включения в БП, реализованный с помощью BPEL, бизнес-правил, для чего имеются инструментальные средства. Привлекательность данного решения определяется следующим. Во-первых, можно разрешить пользователю корректировать БП, а во-вторых, не требуется разработки инструментальных средств. Такой подход обладает существенно большей гибкостью относительно других подходов.

Использование систем, основанных на бизнес-правилах [320] также представляет определенный интерес, однако при реализации распределенной обработки возникают проблемы. Возможен вариант динамической генерации БП, например, средствами семантических веб-сервисов [288]. Однако, это достаточно сложное и не вполне надежное решение. По крайней мере, на сегодняшний день его можно рекомендовать использовать только в полуавтоматическом режиме. Кроме того, учитывая, что процессы ППСМ являются синтезированными процессами, то необходимость в генерации БП не представляется достаточной обоснованной.

Механизмы адаптации могут реализовываться в автоматическом, полуавтоматическом режиме и ручном режиме. Перечисленные механизмы могут использоваться на этапе проектирования и в процессе функционирования.

В таблице 4.7 приведены данные о возможности использования различных аспектов адаптации в ППСМ различных типов (таблица 4.7). В таблице 4.7 в графе “В процессе функционирования” используются следующие обозначения: 1) Конвейерные системы РВ 2) Системы отложенной обработки 3) Локальные ИУС 4) Распределенные ИУС 5) Информационно-управляющие системы.

Таблица 4.7 – Данные о возможности использования различных аспектов адаптации применительно к различным типам ППСМ

№	Аспект	В процессе функционирования					При проектировании
		1	2	3	4	5	
1	Внутренняя адаптация	п	п	п	п	п	П
2	Внешняя адаптация	+	+	+	+	+	+
3	Поведенческая адаптация	х	+	+	+	+	+
4	Структурная адаптация	-	-	х	х	+	+
5	Способность настраиваться на имеющиеся ресурсы	+	+	+	+	+	+
6	Рефлексивная адаптация	х	+	+	+	+	х
7	Самонастройка	х	+	+	+	+	х
8	Самовосстановление	-	х	х	х	+	х
9	Самозащита	-	-	-	-	+	-
10	Самооптимизация	-	+	+	+	+	х

Для обозначения возможности использования некоторого аспекта адаптации применительно к одному из типов ППСМ используется: + целесообразно применять, -- нецелесообразно, х – целесообразность вызывает сомнение, п- реализуется средствами платформы.

Для возможности адаптации ППСМ требуется наличие в платформе многих компонентов, особенно компонентов обработки данных. Вариативность таких компонентов во многом определяет гибкость процессов ППСМ и системы в целом. Компоненты могут объединяться в группы, например, по их функциональному назначению. Такие группы составляют следующие компоненты:

- компоненты для работы с данными (для работы с файловыми хранилищами, с не реляционными БД, с онтологиями предметной области, базами знаний и другие);

- компоненты ядра системы (обеспечивающие информационное взаимодействие подсистем, управление логикой системы, оптимизацию вычислений за счет их параллельного выполнения, интеграцию со сторонними системами, унифицированный доступ к данным предметной области и другие);

- компоненты, необходимые для выполнения обработки результатов измерений (библиотеки алгоритмов обработки и анализа данных, библиотеки описаний процессов, шаблонов и технологий обработки, средства исполнения процессов обработки и другие);

- компоненты, обеспечивающие удобство работы и восприятия информации пользователями, например, компоненты, реализующие технологии ГИС;

- компоненты интеллектуализации обработки (экспертные системы, средства ППР и другие);

- компоненты для управления системой, включая компоненты управления процессами обработки и структурой систем. К первым могут относиться компоненты формирования и коррекции процессов, а также априорной и апостериорной оценки результатов их выполнения. Ко второй - компоненты описания, анализа, оценки и реорганизации структуры системы;

- компоненты обслуживания системы. Примером могут являться компоненты, осуществляющие ведение протокола работы системы, анализа использования системы.

#### **4.6. Выводы по главе 4**

1. Предложен новый класс проблемно- и предметно- ориентированных систем построения моделей объектов. Отличительной особенностью систем является наличие у них собственных моделей, что обеспечивает способность систем к адаптации реализуемых в них процессов к решаемым задачам синтеза, а также способность к самоадаптации. Это позволяет применять новые системы при решении различных прикладных задач, требующих синтеза моделей объектов мониторинга, во многих предметных областях.

2. Предложен метод разработки ППСМ. Метод предусматривает применение гибких методологий при проектировании, создании, использовании и сопровождении систем, т.е. на всех этапах жизненного цикла систем. Разрабатываемые системы описываются в виде совокупности связанных моделей, преобразование которых приводит к преобразованию самих систем. Построение и перестройка таких систем осуществляется по моделям требований. Разработка ППСМ поддержана инструментальными и программными средствами, ориентированными на использование готовых высокоуровневых элементов, включая программные платформы.

3. Предложена новая модель гибкой архитектуры, предусматривающая совместное использование архитектурных и онтологических моделей при описании систем. Использование двух параллельных описаний обеспечивает гибкость систем. Архитектурные модели определяют стандартные элементы архитектурного описания, включающие точки зрения, заинтересованные стороны и другие, расширенные с учетом особенностей ППСМ. При построении архитектурных моделей используется подход, предусматривающий построение стека моделей. Для моделей определены операции трансформации и связывания. Онтологическими моделями обеспечиваются возможности представления и хранения архитектурного знания. Определены способы применения онтологических описаний для всех этапов жизненного цикла систем.

4. Предложен новый метод гибкого проектирования и сопровождения ППСМ, который составляет основу гибкой методологии разработки систем. Метод предусматривает проектирование иерархии систем, включающей системы уровня предметной области, семейств и линеек систем, конечных систем. Результатом процесса проектирования являются модели конечных систем. Обеспечивается возможность построения систем с различными архитектурами. Непосредственно для ППСМ предложена новая трехслойная архитектура, основанная на комбинированном применении веб сервисов и агентов.

5. Предложена новая система архитектурных метрик для ППСМ. Сформулированы основные принципы, в соответствии с которыми определены метрики, приведены требования к метрикам. Для построения системы метрик предложена схема их классификации. Выделены основные группы параметров, которые соответствуют основным группам требований, специфичных для ППСМ: своевременность получения данных, удовлетворенность заинтересованной стороны (сторон) полученными данными, производительность, пропускная способность системы и стоимость. Отдельное внимание уделено метрикам обработки данных.

6. Разработана платформа, обеспечивающая возможность реализации ППСМ на практике. Платформа может рассматриваться как одна из возможных реализаций ППСМ, как фабрика ППСМ и как интеграционная среда. Платформа как фабрика представляет собой мета систему, способную строить множество целевых ППСМ по их моделям. Интеграционные возможности платформы позволяют выполнять интеграцию моделей предметных областей, готовых технологий, сторонних платформ. Для платформы и формируемых на ее основе систем разработано процессно-сервисное представление, которое обеспечивает построение и реализацию гибких процессов при построении моделей объектов по данным мониторинга. Определены механизмы и средства адаптации ППСМ.

В целом, полученные в главе четыре результаты позволили создать новый класс систем с гибкой архитектурой для построения моделей объектов по данным мониторинга, а также предложить новую методологию создания проблемно- и предметно-ориентированных программных систем, предусматривающую их гибкое проектирование, создание и сопровождение.

## Глава 5. Методики и результаты построения моделей объектов по данным мониторинга для решения прикладных задач

### 5.1. Методики построения моделей объектов по данным мониторинга для решения прикладных задач

Методики построения моделей объектов мониторинга включают общую методику построения моделей и частные методики для предметных областей.

Общая методика определяет способы решения задач построения моделей с применением разработанных моделей и методов многоуровневого синтеза [10], [11]. В структурной схеме методики (рисунок 5.1) выделяются три контура, которые обеспечивают синтез формальных моделей, формирование конечных моделей на их основе, построение программных систем, позволяющих реализовать синтезированные модели на практике.



Рисунок 5.1 – Структура общей методики построения моделей объектов

На основе общей методике строятся частные методики для предметных областей. Схема построения методик приведена на рисунок 5.2.



Рисунок 5.2 – Схема построения частных методик для решения прикладных задач

В частных методиках определяются классы задач, решаемых в каждой из предметных областей, и критерии эффективности для оценки результатов. Также в них учитываются модели контента и контекста, характерные для предметных областей, предусматривается применение традиционно используемых в этих областях моделей и методов обработки данных.

Разработанными методиками предусматривается широкое использование моделей и методов статистической и интеллектуальной обработки данных. В дополнение к существующим моделям, методам и алгоритмам разработан ряд новых алгоритмов обработки данных. Среди новых алгоритмов имеются алгоритмы, предназначенные для обработки бинарных потоков [197], [322], временных рядов [198]–[202], [323], а также алгоритмы, ориентированные на решение задач распознавания ситуаций [324], [325]. Сокращенный перечень алгоритмов и описание условий их применения приведены в таблице 5.1. Подробно алгоритмы рассмотрены в [191], [192], [460].

Таблица 5.1 – Перечень алгоритмов интеллектуальной обработки данных

Наименование алгоритма	Условие применения
Алгоритмы сегментации	
Алгоритм сегментации на основе вычисления кривизны функции	Для медленно изменяющихся временных рядов
Алгоритм сегментации на основе вычисления значений спектральной плотности	Для быстро изменяющихся временных рядов
Алгоритм сегментации на основе алгоритма оптимального разбиения	Временной ряд содержит резкие изменения
Алгоритм формирования символьного представления временного ряда	Наличие алфавита
Алгоритмы построения шаблонов	
Базовый алгоритм построения шаблонов	Временной ряд имеет две непрерывных производных; временной ряд имеет разрыв в первой производной
Алгоритм построения шаблонов кусочно-постоянных временных рядов	Временной ряд имеет кусочно-постоянную структуру
Алгоритм построения шаблонов кусочно-линейных временных рядов	Временной ряд имеет кусочно-линейную структуру
Алгоритм построения образов временных рядов измерений	Временной ряд имеет несколько типовых вариантов поведения
Алгоритм построения образов измерений качественных и количественных параметров	Измерения имеют типовые варианты поведения
Алгоритмы сравнения	
Алгоритм нечеткого сравнения временных рядов на основе символьного представления	-
Алгоритм сравнения шаблонов	-
Алгоритм сравнения образов	

Для трансформации данных и их связывания разработана отдельная группа алгоритмов. Среди алгоритмов этой группы имеются алгоритмы поиска мотивов в данных, вычисления интегральных показателей [327]–[330]. Также разработан ряд специализированных алгоритмов, в частности, [331], [332].

### **5.2. Средства для построения моделей объектов по данным мониторинга в прикладных предметных областях**

Для построения моделей объектов разработано программное средство, которое позволяет адаптировать ППСМ для работы с ним специалистов предметных областей. Обобщенная структура предлагаемого средства адаптации ППСМ (СА ППСМ) показана на рисунке 5.3.

Основным элементом СА ППСМ является конструктор запросов (Builder) - средство, позволяющее строить и отлаживать сценарии ППСМ в ручном и (или) автоматическом режиме. Builder предназначен для работы с онтологиями ППСМ и для разработки сценариев построения моделей объектов по данным мониторинга, в том числе, в режиме групповой работы. Builder может быть реализован либо в виде отдельного приложения, либо как модуль расширения для систем проектирования ЯВУ и ООП.

Помимо конструктора запросов в состав средства адаптации ППСМ входят следующие компоненты: ESB (Enterprise Service Bus), подсистема импорта-экспорта, хранилище, в котором в систематизированном виде размещается информация о методах, алгоритмах, сервисах и сценариях (МАСС), правилах и онтологиях, а также графический интерфейс пользователя (GUI). Через подсистему импорта-экспорта имеется выход на стандартные средства ООП и системы разработки ЯВУ. В состав GUI входит редактор онтологий [333], адаптированный для решения задач обработки данных. Для визуализации моделей ППСМ предложены и реализованы новые методы [334], [335]. Для логического вывода могут использоваться стандартные средства, а также новый метод [336], позволяющий выполнять параллельную обработку.

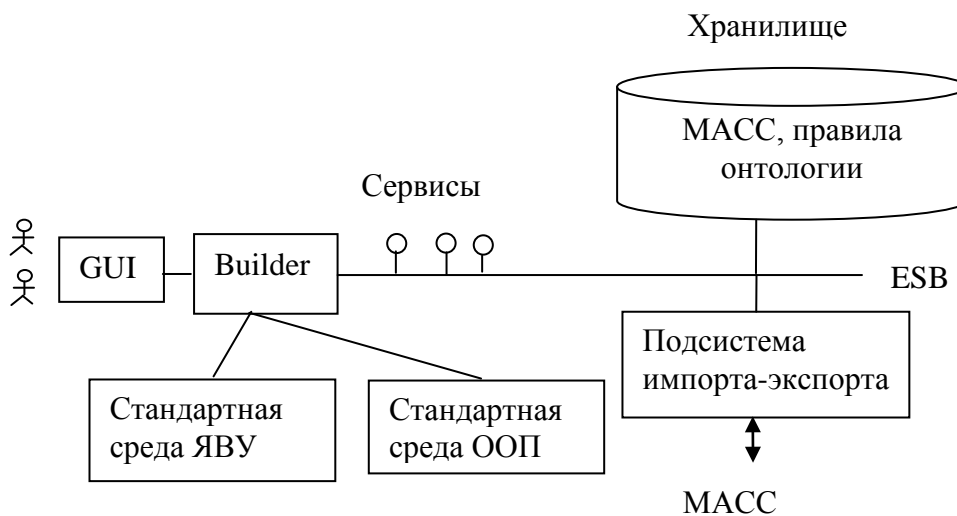


Рисунок 5.3 – Обобщенная структура средства адаптации ППСМ

К основным задачам, которые могут решаться с использованием СА ППСМ, относятся:

- получение информации о доступных МАСС;
- редактирование информации о доступных МАСС;
- построение вручную сценариев в терминах МАСС;
- построение сценариев в терминах МАСС в автоматическом и полуавтоматическом режиме;
- прогон и реконструкция (запоминание) сценариев;



- работа с онтологией.

Builder может поддерживать два режима работы: режим редактирования содержимого хранилища, включая возможность редактирования онтологии, и режим построения сценариев. Проектирование сценариев ведется в терминах функциональных моделей. Результирующие сценарии описываются последовательностью элементарных функций. При построении сценариев адаптивной обработки контента, функции привязываются к уровням модели трансформации. Элементарные функции детализируются до операторов, которые представляют собой отдельные команды или группы команд.

Типовой вариант использования рассмотренного конструктора с привлечением пользователя следующий. Пользователь выбирает из списка доступных операторов (L) необходимый оператор, его настраивает. Если соответствующего оператора нет, то необходимый оператор разрабатывается и его модель помещается в онтологию. Разработка обычно выполняется в отдельном приложении. Когда необходимые операторы созданы и размещены в онтологии, пользователь соединяет операторы между собой, а также с источниками (D) и приемниками (I) данных (рисунок 5.4). Связки операторов могут образовывать сложные последовательно-параллельные структуры.

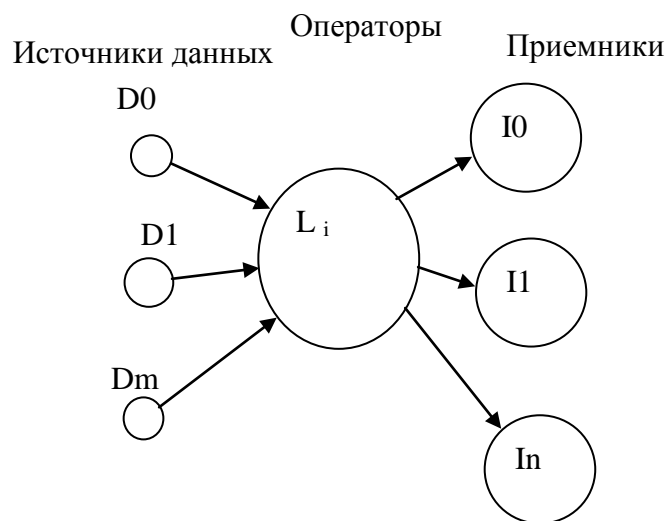


Рисунок 5.4 – Фрагмент сценария контентно адаптивной обработки

Для простоты разработки сценариев может использоваться их табличное представление. Возможны различные стратегии заполнения таких таблиц. Простейшая стратегия – это заполнение информационных моделей, начиная с нижнего уровня. При заполнении нижнего уровня выбираются исходные источники информации. Одним из таких источников может быть некоторое внутреннее или внешнее хранилище данных. Далее определяются данные первого уровня. При этом выполняется поиск операторов, которые могут обеспечить трансформацию данных уровня нуль в данные более высокого

уровня (в рассматриваемом случае уровня один). Если оператор не найден, то он может быть либо синтезирован автоматически на основе описаний имеющихся методов и алгоритмов, либо назначен пользователем вручную. После того, как операторы определены, выполняется один или несколько прогонов сценария и, при необходимости, операторы корректируются. Затем выполняется переход к построению следующего уровня (уровня 1) и процедура повторяется, до тех пор, пока не будет достигнут требуемый уровень. После отладки разработка сценария считается законченной. Вновь созданный сценарий может быть сохранен в хранилище и (или) экспортирован в выбранном формате, например в виде BPEL.

В автоматическом режиме построения сценариев пользователь назначает только исходные источники данных и ожидаемые конечные результаты. Весь процесс построения сценария выполняется автоматически. Затем результирующий сценарий может быть оптимизирован также в автоматическом или в ручном режиме. Для автоматического режима построения сценариев часто применяется механизм семантических веб-сервисов.

### **5.3. Рекомендации по построению моделей объектов по данным мониторинга для решения прикладных задач**

Сформулированы общие рекомендации по построению моделей объектов мониторинга. Они представлены в виде набора типовых решений, достаточных для построения моделей объектов при решении прикладных задач. В состав типовых решений входят: язык для описания синтезируемых программ мониторинга, онтологические модели обработки данных мониторинга, открытая программная платформа построения моделей объектов.

#### **5.3.1. Язык описания программ мониторинга**

Язык описания программ мониторинга предназначен для описания программ, исполняемых на устройствах сбора данных [337]. К возможностям языка предъявлялись следующие требования. Язык должен позволять настраивать параметры фильтрации и агрегации данных, определять условия выполнения команд, определять состав сообщений, которые могут быть отправлены с устройств сбора данных, а также задавать команды на выполнение действий, поддерживаемые устройствами.

Предусмотрена трансляция программ в бинарный формат. Трансляция обеспечивается средствами разработанного компилятора. В результате трансляции формируются программные модули. При трансляции компилятор проверяет синтаксис текста программ, связанность имен, совместимость типов.

При описании программ в бинарном формате используются инструкции универсального императивного языка, что дает возможность расширять разработанный язык. Инструкции являются строго типизированными, это позволяет достичь высокой скорости исполнения программ. Компактность бинарных программ обеспечивается за счет использования языковых конструкций максимально приближенных к решаемым задачам построения моделей объектов, а также за счет компактности самих конструкций.

Программы мониторинга исполняются средствами виртуальных машин, которые устанавливаются на устройствах. Загрузка и выгрузка модулей, их регистрация осуществляется через программный интерфейс, который предоставляется виртуальными машинами. При исполнении кода в виртуальной машине выполняются проверки наличия свободной памяти, переполнения стека, заикливания, выхода индексов за границы массивов, обращения к не проинициализированному полю, деления на ноль и получения остатка от деления на ноль. При наступлении ошибочной ситуации вызвавший ее модуль выгружается, его ресурсы освобождаются, его обработчики перестают воспринимать информацию о внешних событиях.

Загруженным модулям может предоставляться доступ к системным переменным устройств, параметрам, извлекаемым системными функциями, действиям, выполняемым системными функциями.

Для языка описания программ мониторинга определены лексика и синтаксис. Лексика представлена в таблице 5.2, синтаксис приведен на рисунке 5.5.

Таблица 5.2 – Лексика языка описания программ мониторинга

Символы	Описание
пробел , табуляция, \n, \r	разделитель
// или /* */	комментарий
[A-Za-z_][0-9A-Za-z_]*	идентификатор
литеральная строка	" \n \r \t \xNN \\ \0"
блок кода	{ }

Язык описания программ мониторинга типизирован. Типы не преобразуются друг в друга автоматически. Каждое выражение имеет тип, известный на стадии компиляции. Типы не имеют имен. Одинаковыми считаются типы, имеющие одинаковые конструкторы и одинаковые наборы параметров.

В языке поддержаны следующие типы данных: int (32 бита со знаком), bool (false | true), строка (символы в utf-8, адресуемые побайтно), таблица (массив переменного размера, содержащий структуры с полями), ссылка на функцию, массив переменного размера.

Возможно дальнейшее развитие разработанного языка, в частности, поддержка программных интерфейсов, структуры классов, механизмов наследования.

```
program=: 'module' id ';' (global_var_definition | function_definition)* <end of file>
```

```
global_var_definition=: id(name) '=' expression ';'
```

```
function_definition=: id(name) '(' (id(param_name) (',' id(param_name))*)? ')' '{' (statement)* '}'
```

```
statement =: expression ';'
```

```
  | id(local_var) '=' expression ';'
```

```
  | 'if' '(' expression ')' statement ('else' statement)?
```

```
  | (id(label) ':')? 'while' '(' expression ')' statement
```

```
  | (id(label) ':')? 'do' statement 'while' '(' expression ')' ';'
```

```
  | (id(label) ':')? 'for' '(' initializer_list ';' expression ';' (expression (',' expression))*? ')'
```

```
statement
```

```
  | '{' (statement)* '}'
```

```
  | 'break;'
```

```
  | 'return' expression ';'
```

```
expression =: ternary
```

```
ternary =: concatenation ('?' ternary ':' ternary )?
```

```
concatenation = logical ('_' logical)*
```

```
logical =: comparison (('&&' | '||') comparison)*
```

```
comparison =: math (('<' | '<=' | '>' | '>=' | '==' | '!=') math)?
```

```
math =: muls (('+' | '-' muls)*
```

```
muls =: unary (('*' | '/' | '%' | '^' | '&' | '|' | '<<' | '>>') unary)*
```

```
unary =: unary_head unary_tail
```

```
unary_head=: NUMERIC_CONST | STRING_LITERAL
```

```
  | id(var_name)
```

```
  | ('-' | '~' | '!' | '++' | '--') unary_head
```

```
  | '(' expression ')'
```

```
  | '[' id(field_name) (',' id(field_name))* ']' (('*' muls) / ('{ expression (',' expression)* }'))?
```

```
  | '{' expression (',' expression)* '}'
```

```
unary_tail=: '[' expression (':' expression)? ']'
```

```
  | (':'= | '+=' | '-=' | '/=' | '_=' ....) expression
```

```
  | '(' expression (',' expression)* ')'
```

```
  | '.' id(field_name)
```

```
  | '++' | '--'
```

Рисунок 5.5 – Синтаксис языка описания программ мониторинга

Язык разрабатывался и применялся в рамках выполнения проектов в области телекоммуникаций [337]. В материалах проектов имеется более подробное описание языка и результатов его практического применения.

### 5.3.2. Открытая программная платформа построения моделей объектов

Открытая программная платформа представляет собой готовое к использованию решение. Возможности платформы позволяют проводить мониторинг состояния большого числа технических устройств, объединенных в локальные и глобальные сети. Для

наблюдения за отдельными объектами или небольшими группами объектов достаточно применения упрощенного варианта платформы.

Подробное описание платформы, а также реализованных в платформе моделей и методов многоуровневого синтеза приведено в статьях [326], [338]–[340], [483].

Предлагаемая платформа - это распределенная система программные модули которой устанавливаются на конечных устройствах, а также на центральных и локальных серверах существующих систем мониторинга.

Структура платформы показана на рисунке 5.6 [337]. В состав открытой платформы построения моделей объектов (Open Modelling Platform, OMP) входят следующие элементы.

Network segment – сегмент сети. Для функционирования OMP каждая часть сети должна содержать OMP Listener (OMP регистратор событий), а так же AMS (Application Message Service) Cluster (AMS кластер). AMS Cluster позволяет обеспечить обработку данных при большом числе конечных устройств;

OMP Listener – модуль, который взаимодействует с сегментом сети, включающим устройства (Devices), за которыми ведется наблюдение, и AMS Cluster. В каждом сегменте сети устанавливается отдельный OMP Listener (или несколько OMP Listener-ов). Физически OMP Listener может размещаться вместе с компонентами OMP Server (OMP сервер);

AMS Publisher – компонент AMS, отвечающий за размещение файлов на OMP Broadcaster (OMP Поставщик данных), для доставки их на устройства;

AMS DB (Database) – база данных AMS, позволяет OMP получать информацию о наличии устройств в сети, а так же другую информацию об устройствах, имеющуюся в существующих системах мониторинга;

BackEnd Components – набор серверных компонентов OMP;

RabbitMQ – компонент, реализующий очередь сообщений при взаимодействии BackEnd и Listener;

OMP Handler – компонент, который получает сообщения из RabbitMQ, анализирует их и помещает в OMP DB (базу данных OMP);

OMP DB – база данных OMP, в которой размещаются данные о работе сегментов сети;

OMP Server – серверная часть системы OMP;

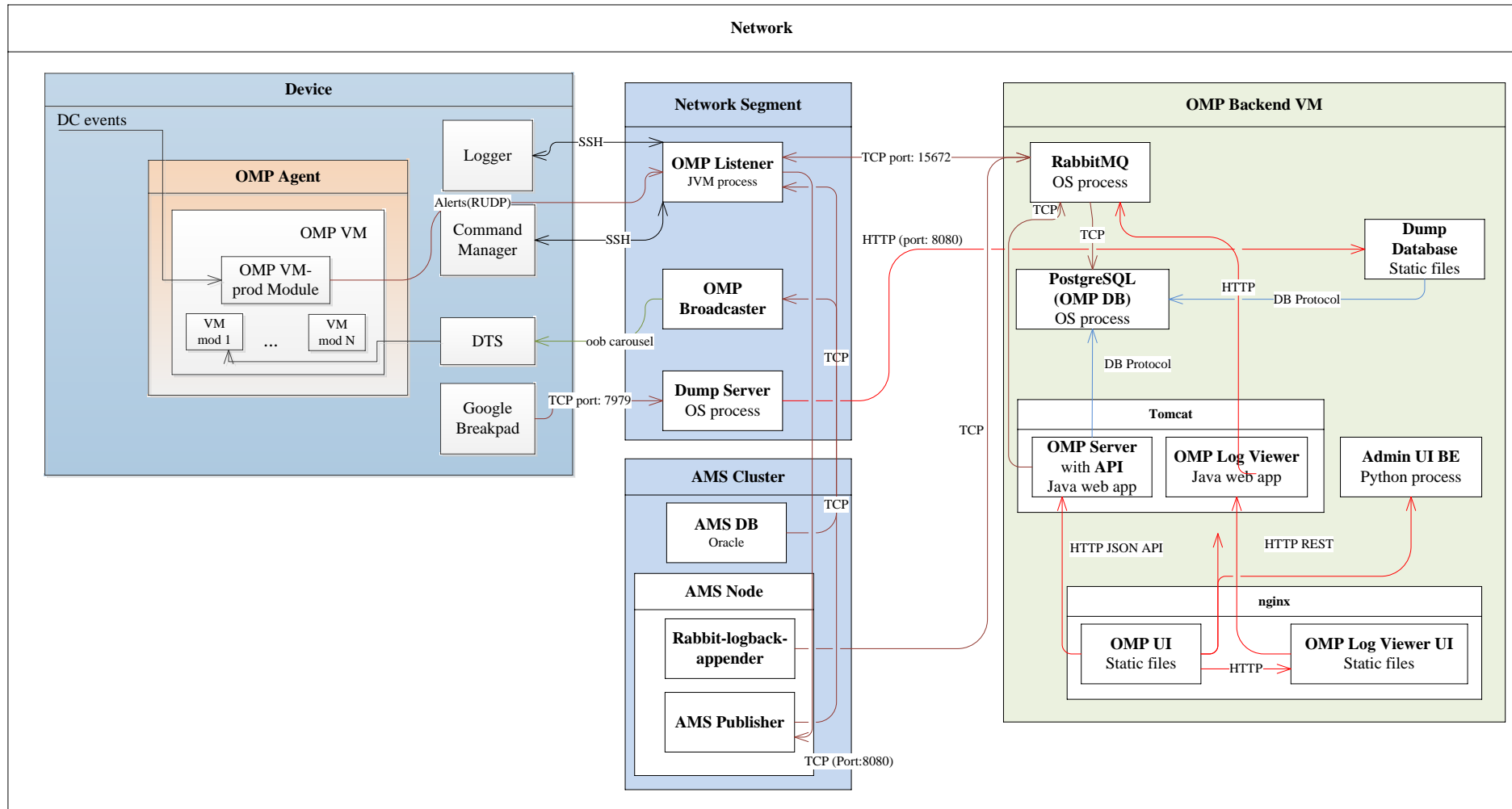


Рисунок 5.6 – Структура программной платформы построения моделей объектов мониторинга

Log viewer – компонент с отдельным пользовательским интерфейсом (User Interface, UI), реализующий обработку журналов работы (лог файлов, Log Files) различных элементов сети, приходящих от AMS (через Listener), и получение из этих журналов информации об устройствах;

Dump Server – процесс, осуществляющий сбор дампов файлов (Dump Files) с устройств. Дамп файлы - это файлы с полным или частичным содержимым памяти устройства или базы данных на момент создания этого файла;

Dump Analyzer – компонент с отдельным UI, реализующий обработку дампов файлов, получаемых от Dump Server.

На конечных устройствах устанавливаются OMP агенты (OMP Agent) [337]. OMP Agent – специальное приложение, устанавливаемое на устройствах и содержащее виртуальную машину (OMP VM (Virtual Machine)), в которой выполняются синтезируемые программы мониторинга. Типовая структура OMP агента показана на рисунке 5.7.

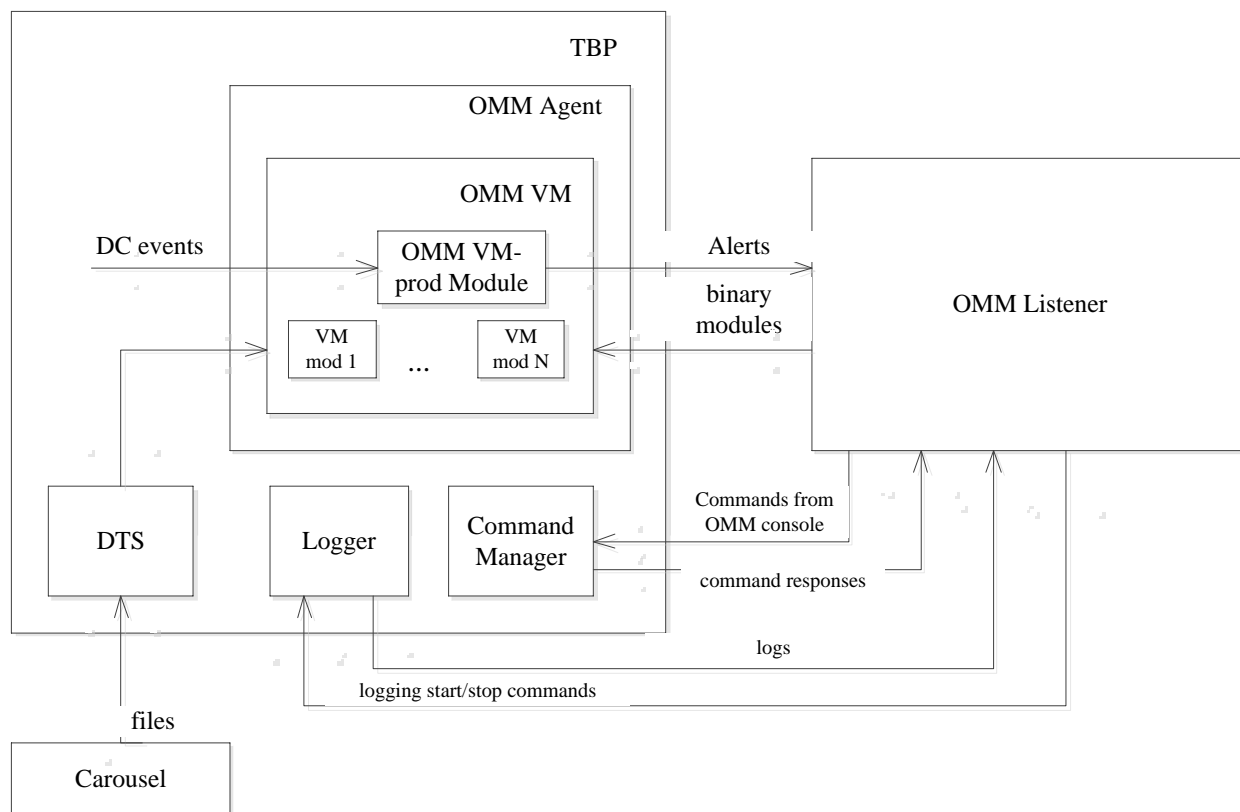


Рисунок 5.7 – Типовая архитектура агента ППСМ

В состав OMP агента входят следующие основные компоненты:

OMP VM – среда выполнения модулей, полученных в результате компиляции программ мониторинга;

OMM VM prod Module– модуль, определяющий базовую логику работы агента;

Logger – часть устройства, отвечающая за сбор и предоставление журналов работы устройств;

Command Manager – компонент, отвечающий за получение и выполнение команд на устройстве.

Для изменения логики работы OMR агента достаточно написать на языке описания программ мониторинга скрипт, определяющий новую логику работы, выполнить его компиляцию и загрузить скомпилированный модуль в виртуальную машину.

#### **5.4. Результаты построения моделей объектов по данным мониторинга в прикладных предметных областях**

Построение моделей объектов проводилось в четырех предметных областях: области телекоммуникаций, области объектов космического назначения, области океанографии, области практической и исследовательской медицины.

Построение моделей выполнялось с применением разработанных моделей и методов на реальных данных. Подробно ход построения моделей и результаты их построения рассмотрены в монографиях автора [192], [341].

Построение моделей в каждой предметной области предусматривало: описание исходных данных, постановку частных задач построения моделей объектов, оценивание полученных результатов.

Состав рассматриваемых задач определялся специалистами предметных областей, обладающими экспертными знаниями, также совместно с ними выполнялось построение моделей объектов. Оценка эффективности предложенных решений проводилась по итогам эксплуатации систем на протяжении нескольких лет.

##### **5.4.1. Результаты построения моделей объектов по данным мониторинга для области телекоммуникаций**

Построение моделей проводилось по запросу операторов сетей кабельного телевидения США и Северной Канады в рамках проектов, реализуемых компанией [337]. Потребность в построении моделей была вызвана частыми отказами, возникающими в этих сетях, сложностью их локализации и устранения. Существующие сети включают в состав большое число технических устройств старых образцов, используют имеющиеся сети передачи данных, которые имеют многие технические ограничения и низкую надежность. В ходе создания ППСМ для операторов кабельного телевидения был разработан ряд новых моделей и методов, а также архитектурных и программных решений [99], [164], [165], [218], [265], [278].



Построение моделей проводилось по данным нескольких сегментов сетей, общее число наблюдаемых объектов в сети составляло в различные периоды от нескольких сотен до нескольких миллионов. Методика построения моделей, а также полученные результаты построения моделей приведены в монографии [192].

Построение моделей было направлено на решение следующих прикладных задач:

- мониторинг состояния сети кабельного телевидения при различных условиях эксплуатации, в том числе в условиях высоких нагрузок;
- выявление и локализация ошибочных ситуаций, возникающих в результате действий пользователей или неисправности технических средств;
- прогнозирование и предотвращение возникновения сбоев в работе сети;
- формирование рекомендаций по устранению выявленных ошибочных ситуаций.

*Состав исходных данных для построения моделей.* Основными источниками данных о состоянии сетей являлись:

1. Сообщения об ошибках телевизионных приемников (ресиверов) (ТВР), установленных у конечных пользователей. Сообщения относятся к оперативным данным. Они содержат значения отдельных параметров ТВР, оповещающих о возможности возникновения проблемной ситуации. Часть параметров может обрабатываться на уровне программного кода. Как правило, обработка сводится к выводу на экран пользовательского сообщения об ошибке и/или отправке сообщения на сервер. Информация об ошибках передается на сервер сразу после их возникновения или с минимальной задержкой, определяемой логикой работы ТВР. В качестве примеров ошибок можно привести: «Отсутствует программа передач» (из-за отсутствия программы передач пользователь не может выбрать нужный канал), «Черный экран» (нет видеосигнала, пользователь видит только черный экран), «Перезагрузка ТВР», (произошла внезапная перезагрузка ресивера).

2. Метрики о состоянии аппаратных средств серверного оборудования или его окружения. Для сбора таких метрик разрабатываются отдельные сервисы. Эти сервисы могут предоставлять параметры загрузки процессоров центрального или локальных серверов, размер занятой и свободной памяти, данные о наличии свободного места на конкретном жестком диске или его разделе, объем трафика сети и другие. К сервисам могут формироваться запросы, включающие запросы о разделе, где хранятся полученные лог файлы, о количестве обращений в секунду к некоторому системному или пользовательскому сервису, об общем количестве запросов на некотором интервале времени, о времени выполнения запросов, об ошибках, возникших в процессе выполнения запросов по типам сервисов, в том числе с распределением по кластерам.

3. Лог файлы пользовательских и серверных устройств. Лог файлы представляют собой системные журналы работы технического средства. Буферные лог файлы выводятся через буфер в UDP или TCP порты. Исторические лог файлы – это файлы с наиболее важными лог записями, которые сохраняются на ТВР. Их можно получать по запросу. Возможно определить логические правила, по которым лог файлы будут выгружаться с устройств, например, “привязать” получение лог файлов к некоторым событиям. Поддерживаются несколько уровней логирования. В зависимости от уровня в лог файлы может выводиться информация об ошибках, информационные сообщения, отладочные сообщения и т.д. Такие файлы могут формироваться как на стороне ТВР, так и на стороне сервера, или другого пользовательского устройства. Файлы с логами могут храниться на центральном сервере или на выделенных серверах в сыром или в обработанном виде.

4. Краш-дампы (пакеты данных специальных форматов), которые генерируются ТВР в момент фатального отказа, после которого требуется перезагрузка устройства. Краш-дампы могут отсылаться на сервер автоматически, либо оператор может удаленно забирать дампы непосредственно с ТВР. Дампы хранятся и обрабатываются аналогично буферным логам, в виде файлов.

5. Телеметрические данные - множество параметров, характеризующих состояние пользовательского устройства. Такие данные могут передаваться по запросу или выгружаться на сервер по расписанию. Телеметрическими данными являются:

а) одиночные системные сообщения (Heartbeats), которые могут посылаться с ресиверов на сервер с заданной частотой и использоваться для определения текущего состояния ТВР;

б) периодически отправляемые конфигурируемые наборы системных параметров;

в) данные о поведении пользователей – информация о просматриваемых каналах, длительности пребывания пользователей на канале, составе дополнительных запросов пользователей и т.д.;

г) SNMP-параметры, получаемые специальным сервисом путем разовых или периодических опросов ТВР. Данные передаются по SNMP-протоколу [342], разработанному для удаленного получения диагностических параметров и управления пользовательскими устройствами. Примеры SNMP-параметров: общий объем памяти ТВР и объем свободной памяти, объем флеш памяти, объем графической памяти и т.д. Общее число таких параметров, как правило, составляет несколько сотен.

6. Данные об управлении пользовательским оборудованием через глобальную сеть. Такие данные представляют собой нотификации, сформированные в соответствии со стандартом TR-069 [343]. Используемый для передачи нотификаций протокол CWMP

позволяет выполнять следующие действия: начальная настройка устройства при его загрузке; внесение изменений в настройки уже работающего устройства. Такие нотификации на практике используются редко. Это связано с тем, что поддержка протокола требует затрат значительных дополнительных ресурсов. Вместо нотификаций широко используется рассылка широковещательных сообщений через карусели на группы ТВР или специализированные частные интерфейсы, позволяющие выполнять отправку команд на отдельные устройства.

7. Сообщения о произведенных действиях «самолечения» на ТВР. Такие процедуры, как правило, определяются логическими правилами, их выполнение может сопровождаться дополнительной информацией, в частности, формированием лог файлов. Примеры действий самолечения: освободить память, остановить / запустить сервис или приложение, произвести перезагрузку ресивера.

8. Диагностические страницы ТВР. Это перечень параметров о состоянии устройства, который выводится пользователю на экран телевизора. Они организованы в виде последовательности страниц. Те же самые страницы могут быть переданы на сервер. Они отсылаются в текстовом виде, по запросу. Можно задать периодическую автоматическую доставку параметров.

*Состав синтезированных моделей сетей кабельного телевидения.* В состав синтезированных моделей телекоммуникационных сетей с применением предложенных методов входили:

- модели сети в целом и ее сегментов различных уровней. Состояние и поведение сети определяются через состояния и поведение ее элементов – сетей передачи данных, серверного оборудования, пользовательских устройств. Определяются интегральные показатели для оценки состояния сетей и динамики их изменения, устанавливаются взаимосвязи между показателями;

- модели пользовательских устройств. Пользовательские устройства характеризуются параметрами технических средств, а также параметрами установленных на них программных средств. Между параметрами и их группами устанавливаются зависимости;

- модели ошибок по типам. Для различных типов ошибок определены способы их проявления, возможные причины возникновения, варианты устранения. Установлены параметры, изменение значений которых может свидетельствовать о возникновении ошибки одного из типов;

- модели пользователей. Модели построены по данным о поведении пользователей. Модели основаны на результатах анализа популярности каналов по данным за десяти

часовые временные интервалы и за суточные интервалы. Также при построении моделей исследовалось поведение отдельных пользователей и групп пользователей.

Построенная модель телекоммуникационной сети подробно рассмотрена в [218]. Модель реализована в виде онтологии.

Построены следующие модели зависимостей:

1. Модели зависимостей между параметрами элементов сети - ТВР и серверными компонентами - при возникновении различных ошибок. Исследовались параметры ТВР и серверных компонентов, а также записи лог файлов. Построенные зависимости включают описания ошибок, причин их возникновения, характерные значения параметров сервера и ТВР, а также соответствующие записи лог файлов.

2. Модели зависимостей между ошибками различных типов. Исследовалась зависимость распределения типов ошибок от времени их регистрации на устройствах. Информация была получена из лог файлов сервера, БД ошибок ТВР.

3. Модели зависимостей для ошибок одного типа. Для определения зависимостей были построены следующие распределения: количества событий по типам событий; количества событий по времени их возникновения с учетом типов ошибок; событий по пользователям с учетом типов ошибок; событий по пользователям с учетом уровня сообщений (Warn, Error), выводимых в лог файлы; распределение сообщений лог файлов по их уровням; сообщений лог файлов по типам ошибок; событий по компонентам и процессам. Зависимости описывались с использованием регрессионных моделей.

4. Модель временных (условно постоянных) зависимостей ошибок серверного оборудования. Исследовались распределения различных типов ошибок серверных компонентов для различных сегментов сети при различных условиях. Обработка данных проводилась с привязкой к суточным интервалам. Модели представляют собой гистограммы распределения частот возникновения событий различных типов по часам.

5. Модель зависимостей поведения компонентов ТВР. Зависимости описывались в виде шаблонов поведения компонентов программных платформ ТВР. Шаблоны представляют собой последовательность событий, регистрируемых в лог файлах. Построены шаблоны, описывающие типовое поведение компонентов, наблюдаемое при выполнении различных пользовательских сценариев, а так же поведение при возникновении ошибок.

Модели описывают ТВР на различных уровнях – уровне программных платформ, уровне отдельных компонентов, уровне модулей, уровне значений параметров.

*Состав синтезированных процессов сбора и обработки данных о состоянии сетей.* Синтезировались следующие процессы: регулярные процессы мониторинга, которые

выполнялись на стороне ТВР и серверной стороне, а также специализированные процессы для локализации возникших ошибочных ситуаций.

Процессы регулярного мониторинга предусматривают контроль параметров, связанных с отслеживаемыми проблемными ситуациями. Процедуры мониторинга генерируются сервером, выполняться они могут как на стороне ТВР, так и на стороне сервера. Процедуры, как правило, предусматривают оперативное выявление и обработку только наиболее часто встречающихся ошибок. К таким ошибкам относятся ошибка отсутствия изображения на экране телевизора, отсутствие данных, поставляемых сторонними сервисами, отсутствие программы передач, не рабочее состояние части каналов, отсутствие возможности работы с данными на устройстве (записи или чтения). В соответствии со статистическими данными перечисленные ошибки составляют порядка 95 % всех ошибок, оказывающих негативное влияние на пользовательские функции ТВР. Отдельную группу ошибок составляют ошибки, связанные с незапланированными перезагрузками пользовательских устройств.

Процедура локализации и устранения ошибок направлена на выявление причин их возникновения. Одна и та же ошибка в зависимости от условий может быть вызвана различными причинами, она может оказывать различное влияние на пользователей, также, как правило, отличаются и пути ее устранения. Возможными причинами возникновения ошибок могут являться: превышение допустимого времени ожидания отклика от внешних систем, ошибки в полученных данных, например, программе передач, проблема авторизации, неисправное состояние устройства приема и передачи данных. Для локализации ошибок синтезировались процессы, обеспечивающие многократное последовательное уточнение параметров состояния элементов сети, исходя из значений ранее запрошенных параметров.

В соответствии с процессами были синтезированы программы сбора и обработки данных, предназначенные для выполнения на ТВР.

При наличии достаточных ресурсов на обработку лог файлов синтезировались процессы для поиска в них шаблонов, соответствующих ошибкам загрузки данных, нарушению целостности данных, отмене запросов или повторных циклических перезапросов.

*Оценки результатов построения моделей.* При построении моделей в зависимости от условий выполнялся поиск целесообразной конфигурации ППСМ. Конфигурации определялись характеристиками технических и программных средств, сетей передачи данных, параметрами процессов сбора и обработки данных.

К параметрам технических средств относятся параметры сервера, параметры сетей передачи данных, параметры моделей ТВР и другие. В составе параметров сервера

рассматривались: оперативная память (ОЗУ), модель и характеристики ЦП, включая число ядер, скорость чтения, интерфейс и объем жесткого диска, тип, пропускная способность и разъем сетевого адаптера, тип и версия операционной системы. Типовыми параметрами сетей передачи данных являются: топология сети, тип сети. Могут использоваться сети различных типов, включая Aloha, DOCSIS, Intranet (IP-protocol). В первую очередь, они отличаются скоростью передачи данных.

Параметры ТВР определяются ее типом. Типы отличаются моделью ТВР, количеством тюнеров, скоростью исполнения инструкций, скоростью CPU (МГц), общим объемом ОЗУ (Мбайт), объемом памяти, доступном ядру Linux (Мбайт), объемом свободной памяти работающей ТВР (Мбайт), общим объемом флэш накопителя (Мбайт), свободным объемом флэш накопителя после старта ТВР (Мбайт), общим объемом энергонезависимой памяти (NVM) (Кбайт), доступной NVM (Кбайт), общим объемом жесткого диска (HDD) (Гбайт), объемом свободной памяти HDD, версией ядра Linux и другими параметрами.

К характеристикам программных средств, оказывающих существенное влияние на параметры, относятся характеристики серверных кластеров, характеристики используемых протоколов передачи данных. Возможности кластеров определяются технологиями, на основе которых они построены. В рассматриваемой системе для построения кластера использовались технологии Hazelcast [344].

При взаимодействии элементов в сети могут использоваться протокол UDP (сообщения без подтверждения и без предварительной установки канала) или RUDP (сообщения с подтверждением доставки).

При конфигурировании ППСМ определялось число физических и логических серверов, на которых разворачивались компоненты системы, а также параметры процессов сбора и обработки данных. В частности, определялось число разворачиваемых компонентов сбора и обработки данных ППСМ, настраивались следующие логические параметры: временной интервал распределения отправки сообщений от ТВР к серверу; частота опроса ТВР; уровень детализации сообщений, выводимых в лог файлы; уровень агрегирования данных о состоянии ТВР; уровень агрегирования данных о состоянии сервера; способы агрегирования данных для различных групп ТВР; распределение логики обработки между ТВР и серверной частью; состав отслеживаемых событий на ТВР, на сервере; состав правил, запускающих выполнение действий на ТВР; необходимость в регулярном мониторинге, реализуемом на ТВР, способ его выполнения (конфигурация резидентного модуля мониторинга); распределение задач обработки между центральным сервером и локальными серверами; тип и объем используемых хранилищ данных или файлов на серверах, политика их очистки; состав и способы коммуникации со сторонними компонентами и системами;

способы отправки управляющих команд на ТВР (широковещательный, одноадресный, многоадресный); группировка ТВР (группировка в соответствии с логической сетевой топологией, возможностями аппаратного и программного обеспечения, предпочтениями пользователей и другие); способы оценки загруженности сети (сетевое трафика) перед отсылкой больших объемов данных с ТВР; состав системных событий, на которые должны подписываться компоненты ТВР; состав и объем регулярно отправляемых на сервер данных (отчеты об использовании памяти ТВР, отчеты о производительности ТВР, отчеты о состоянии сети и другие).

Конфигурация ППСМ определяет функциональность систем и число устройств, за которыми может вестись наблюдение.

Применение сконфигурированных систем в нескольких сегментах крупных сетей, включающих несколько десятков тысяч пользовательских устройств различных типов, позволило установить следующее: своевременно зарегистрированы и локализованы 98,5% всех ошибок, из них заблаговременно были выявлены более 70% ошибок. В результате, применение разработанных методов позволило сократить период неисправного состояния пользовательских устройств до нескольких минут. Ранее для устранения неисправности к пользователю направлялся специалист службы поддержки или службой поддержки проводилась удаленная диагностика и восстановление работоспособности устройства. При этом устранение неисправности могло занимать несколько часов, что вызывало у пользователей значительное недовольство. Применение ППСМ обеспечило сокращение времени выявления и локализации ошибок до секунд, в отдельных случаях, до минут. Кроме того, оказалось возможным более, чем в двое сократить число специалистов службы поддержки.

#### **5.4.2. Результаты построения моделей объектов по данным мониторинга для области объектов космического назначения**

ППСМ разрабатывались для использования на космодромах и полигонах. Построение моделей объектов космического назначения (ОКН) и связанных с ними объектов велось в интересах решения задач инженеров-анализаторов на этапе эксплуатации ОКН. Работы по созданию ППСМ данного типа выполнялись в [345], [346] по государственным заказам [347]–[362] с 2004 года. Разработанные за этот период прикладные модели, методы и алгоритмы, а также программные системы нашли отражение в статьях [195]–[202], [216], [247], [322], [323], [333], [363]–[405].

Методика построения моделей объектов космического назначения, предусматривающая использование моделей и методов многоуровневого синтеза, а также полученные результаты построения моделей приведены в монографии [341].

Представленные в [341] результаты получены по данным технологического обслуживания и данным пусков РКН семейства «Союз», выполненных с космодрома Плесецк.

В области объектов космического назначения построение моделей проводилось в интересах решения следующих групп прикладных задач.

Первая группа задач формулировалась специалистами центра контроля параметров (КП) зарубежных образцов технических объектов на основе данных, передаваемых американской стороной в рамках договора СНВ. Задачи, решаемые центром КП, состоят в восстановлении и анализе летно-технических (ЛТХ) и баллистико-навигационных (БНО) характеристик зарубежных космических объектов по результатам анализа содержимого информационных потоков, передаваемых с них во время пусков [195]–[197], [322], [368], [369], [374], [387], [392], [400].

Вторая группа задач решалась при реализации серии государственных заказов на разработку систем поддержки деятельности подразделений, обеспечивающих комплексную подготовку к пускам и проведение пусков отечественных образцов космической техники, а также подразделений, ответственных за анализ результатов пусков. Основной задачей таких подразделений является анализ состояния ОКН по результатам измерений их параметров [378], [380], [381], [389], [404], [405].

Третью группу задач определяли задачи анализа надежности, безопасности, выявления причин и факторов, приводящих к отказам, инцидентам, авариям, катастрофам, задачи планирования профилактики и предотвращения происшествий на объектах РКТ и наземно-космической инфраструктуры (НКИ). Эти задачи решаются рядом подразделений, среди которых служба вооружения космодрома, центры испытания и применения космических средств и др. [376], [394], [397].

Отдельного внимания заслуживает опыт применения ППСМ для решения нескольких автономных задач. Наиболее интересными являются задача информационной поддержки национальных технических средств контроля (НТСК) испытаний технических средств в соответствии с международными договорами, в частности, договором СНВ-III между РФ и США, и задача оперативного координационного планирования применения средств наземного комплекса управления космическими аппаратами.

Суть задачи информационной поддержки НТСК состояла в построении программ для проведения циклов измерений (ЦИ), позволяющих получить максимально возможный объем информации о наблюдаемых технических средствах. Построение программ выполнялось при высокой неопределенности условий решения задачи. Как правило, имелась информация только о типе объекта РКТ, планируемого к пуску, и районе пуска [398], [401].



Задача координационного оперативного планирования состоит в назначении сеансов связи средств наземного комплекса управления (НКУ) с космическими аппаратами (КА). За счет сеансов связи реализуются регламентированные циклы управления КА, обеспечивается использование КА по целевому назначению [406].

*Состав исходных данных для построения моделей.* К основным типам данных области объектов космической техники относятся:

- структурированные бинарные потоки телеметрической информации, передаваемые с объектов КТ. При формировании потоков могут учитываться существующие стандарты. Одним из распространенных стандартов является IRIG 106 [407]. Потоки содержат результаты измерений параметров объектов, выполненных с помощью установленных на объекте датчиков;

- диаграммы качества информации. Диаграммы показывают качество зарегистрированных потоков телеметрической информации, которое определяется объемом потерянной информации. Регистрация потоков проводится измерительными пунктами, размещенными в различных районах и регионах;

- результаты измерений значений параметров. Результаты измерений могут представлять собой временные ряды, а также отдельные количественные или качественные значения. Значения параметров могут быть соотнесены со шкалой времени или шкалой событий, иметь пространственную привязку;

- параметры технических объектов (летно-технические, баллистико-навигационные характеристики и другие). Значительная часть характеристик рассчитывается с использованием специализированных методик. Как правило, методики разрабатываются предприятиями изготовителями технических объектов;

- интегральные расчетные характеристики состояния технических объектов (показатели надежности, безопасности и другие). Для расчета интегральных характеристик имеется значительное число методик, к разработке которых привлекаются как специалисты предприятий изготовителей, так и представители сторонних организаций, специализирующиеся на вопросах надежности и безопасности сложных технических объектов;

- технологические процессы, реализуемые при комплексной подготовке объектов КТ к пускам, а также процессы планирования мероприятий, связанных с подготовкой и проведением пусков;

- технические характеристики объектов. Технические характеристики объектов определяют ограничения на возможности наблюдения за объектами, а также взаимодействия

объектов. Учет технических характеристик необходим при планировании сеансов связи средств НКУ с КА, а также при построении программ для НТСК.

*Состав синтезированных моделей объектов космического назначения.* С применением предложенных методов были синтезированы следующие модели:

- модели потоков телеметрической информации. Модели отражают структуру телеметрических потоков и их содержание. В рамках одного потока может происходить смена его структуры и содержания;

- модели отдельных параметров объектов. Построение моделей выполнялось для параметров различных типов: константных, динамических, сигнальных. Для константных параметров были определены ожидаемые значения параметров и выявлены их возможные отклонения для различных условий. Модели динамических параметров описывают их поведение в виде векторов характеристик, рассчитанных на стационарных интервалах. Для описания параметров были построены признаковые пространства. Для сигнальных параметров определялись их состояния и моменты смены состояний;

- модели объекта в целом и его элементов. Модели объекта и его элементов строились на основе результатов измерений значений отдельных параметров. Модели отражают зависимости между значениями параметров элементов объекта и изменения этих зависимостей в пространстве и во времени. Строились модели, учитывающие физическую структуру объекта и его элементов, а также модели, отражающие его логическую структуру;

- модели возникновения нештатных ситуаций и их развития. Построенные модели отражали параметры, в значениях которых наблюдались отклонения, моменты времени, с которых отклонения начали наблюдаться. Выполнялась синхронизация моделей по различным шкалам. Модели строились на основе сравнительного анализа данных однотипных конструктивных элементов объекта, а также с учетом данных предшествующих пусков;

- модели технологических циклов работы с объектами КТ. Модели отражают предусмотренные технологическими циклами последовательности команд, временные интервалы их прохождения, результаты контроля выполнения команд. Модели позволяют выявлять случаи опережения и запаздывания прохождения команд;

- модели качества работы измерительных средств и измерительных пунктов. Модели строились на основе сравнительной оценки объемов теряемой информации при регистрации телеметрических потоков различными станциями в различных условиях на одном измерительном пункте и на многих.

*Состав синтезированных процессов сбора и обработки данных об объектах космического назначения.* При построении моделей синтезировались следующие процессы:

- процесс восстановления структуры исходных потоков данных. Построенные процессы предусматривали вычисление корреляционных функций, построение частотно-ранговых распределений и их аппроксимацию, а также оценку полученных результатов на основе поиска служебных слов;

- процесс восстановления значений измеряемых параметров объектов. Процесс предусматривал преобразование информационных единиц бинарного потока в количественные значения. При этом определялись тип представления значений параметров (знаковое / без знаковое), тип параметров (константный / сигнальный (контактно-кодовый) / счетчик / меандр / мантисса / порядок);

- процесс выявления стандартных зависимостей. Процессом обеспечивалось выявление множества характерных зависимостей, как между отдельными параметрами, так и их группами (синусно-косинусные зависимости / интегрально-дифференциальные пары / элементы матриц пересчета координат);

- процесс построения алгоритмов обработки. Процессом выполнялось построение новых алгоритмов обработки за счет совместного использования существующих. Информация, необходимая для построения алгоритмов «добывалась» с использованием процедур разведочного анализа.

*Оценки результатов построения моделей.* При построении моделей использовались разработанные ППСМ для объектов космической техники.

Основными количественными показателями построения моделей объектов являются время и точность обработки. Такие оценки представляют интерес, в первую очередь, при построении моделей потоков телеметрической информации и определении типов параметров. Это связано с большим объемом ГТС, сложностью их структуры, сложностью поведения параметров, плохим качеством данных и т. д. Для получения оценок были построены потоки с различной структурой, отличающиеся по числу и составу параметров, уровню шума и т. д. При их построении учитывались требования стандарта IRIG. Также были обработаны потоки, полученные в результате проведения реальных пусков на космодроме «Плесецк». Обработка реальных данных проводилась инженерами-анализаторами. Общее число обработанных потоков составило несколько тысяч. Время обработки потоков находилось в диапазоне от нескольких секунд до нескольких минут. Время обработки одного ГТС не превышало 8 мин. Это в десятки раз меньше, чем время работы ранее применявшихся алгоритмов.

Оценки точности обработки определялись числом и составом коррекций, которые были внесены экспертами в восстановленные структуры. Полученные таким способом оценки показали, что построенные модели отражают реальную структуру и содержание

потоков более, чем в 92% случаев. За счет дообучения алгоритмов восстановления структуры этот показатель может быть увеличен.

Результаты обработки данных при комплексной подготовке к пускам показали, что ППСМ обеспечивают достоверное решение 85%-95% прикладных задач, точность обработки составляет порядка 98%.

Еще одним показателем является стоимость создания и сопровождения систем. Этот показатель оценивается на основе объема работ, выполненных при разработке систем, и также числа запросов на модификацию систем, полученных от потребителей. Стоимость создания систем была сокращена более чем в два раза, стоимость сопровождения систем стала требовать незначительных ресурсов.

Также были получены косвенные оценки востребованности систем предметными специалистами. Рассматривались следующие показатели:

– объем работ, выполненных специалистами с использованием развернутых ППСМ. Данные, собранные системами о работе пользователей, показывают, что системы активно применяются при решении 85 - 95 % прикладных задач;

– число запросов на увеличение количества рабочих мест и создание новых систем для решения других групп задач. За пятилетний период эксплуатации систем число рабочих мест увеличилось от нескольких единиц до нескольких сотен;

– отзывы специалистов. На объектах эксплуатации все отзывы фиксируются. На основе негативных отзывов могут формироваться рекламации. За период эксплуатации систем рекламаций получено не было.

#### **5.4.3. Результаты построения моделей объектов по данным мониторинга для области исследовательской и практической медицины**

Задачи построения моделей в области исследовательской и практической медицины решались на базе Северо-Западного Национального медицинского исследовательского центра им. В. А. Алмазова Минздрава России (СЗНМИЦ им. В. А. Алмазова) (Центр) [408] с участием специалистов нескольких разноплановых подразделений Центра. ППСМ создавались с целью использования совместно с существующими медицинскими информационными системами (МИС), которые, в основном, ориентированы на сбор данных и поддержку документооборота. Возможности по обработке данных у развернутых в Центре МИС крайне ограничены.

Построение моделей проводилось по данным о кардиологических больных, накопленных в БД МИС более, чем за десятилетний период. Результаты нашли отражение в статьях [162], [173], [174], [193], [217], [327]–[330], [409]–[414].

Подробное описание методики построения моделей и полученных результатов приведено в [341].

Построение моделей было направлено на решение следующих прикладных задач:

- комплексная оценка состояния пациентов по результатам измерений параметров состояния организма с использованием различных диагностических средств. При комплексной оценке организм пациента рассматривается как единая система. Требовалось выявлять, насколько изменение одного параметра сопровождается изменениями связанных с ним других параметров [162], [173], [328]–[330];

- совместный анализ объективных данных о пациентах, получаемых от диагностических средств, и записей врачей [174], [409], [410];

- выбор диагностических процедур. Требовалось определить минимальный набор диагностических процедур, который может позволить достаточно точно установить диагноз с учетом возможностей современных диагностических средств;

- выявление медицинских знаний на основе обработки данных, накопленных в БД МИС. При этом требовалось решение задач восстановления скрытых взаимосвязей между параметрами различных систем организма, зависимостей состояния организма от параметров внешней среды и других зависимостей, их формализованное описание [327];

- структурирование и систематизация медицинских данных. Требовалось построение доменных моделей и их наполнение за счет конвертации данных из существующих БД. Доменные модели представляют собой специализированные словари, которые определяют медицинские термины для некоторого подразделения или медицинского учреждения [413].

*Состав исходных данных для построения моделей.* Медицинские данные представляют собой большие массивы разнородных неструктурированных или слабоструктурированных объективных и экспертных данных.

Объективными считаются результаты измерений, полученные с применением различных диагностических средств. Число параметров, измеряемых за одни сутки для одного пациента, исчисляется уже сотнями и продолжает расти. Данные, получаемые от различных диагностических средств, могут существенно отличаться как по структуре, так и по формату. Структура выходных данных многих средств формализована только частично.

К экспертным данным относятся записи лечащих врачей, профильных специалистов и других медицинских работников. Объем одной истории болезни составляет десятки страниц. Записи врачей представляют собой специализированные тексты на естественном языке.

Измерения параметров и записи врачей, в общем случае, носят периодический характер. Относительно регулярно измеряются только такие параметры, как давление, температура и т. п.

*Состав синтезированных моделей.* По медицинским данным выполнен синтез следующих моделей:

- динамической связанной модели параметров организма. Модель имеет иерархическую графовую структуру. Она содержит информацию об исходных параметрах, выявленных зависимостях между ними, интегральных показателях, вычисленных для различных групп параметров, а также зависимостях, установленных между интегральными показателями [415]. Модель позволяет оценивать состояние организма в целом и динамику его изменения;

- доменных моделей. Доменные модели строятся на основе результатов анализа медицинских текстов [416]. Термины доменных моделей связываются с терминами, определенными в стандартах или в других словарях;

- моделей диагностических исследований. Модели определяют условия, при которых диагностические исследования могут позволить подтвердить или опровергнуть предварительный диагноз, уточнить поставленный диагноз. В частности, показано, что ЭХОКГ позволяет подтвердить диагнозы, относящиеся к группам I20–I25 МКБ-10 [417].

*Состав синтезированных процессов сбора и обработки медицинских данных.* Для обработки медицинских данных синтезировались следующие процессы:

- процессы формализации, структурирования и согласования данных о состоянии пациентов, полученные из различных источников, в том числе, данные предоставляемые существующими МИС. Структурирование данных предусматривало восстановление и согласование событийных и временных шкал. При согласовании выполнялось сопоставление моделей, построенных по данным из различных источников;

- процессы обработки количественных медицинских данных. Процессами обработки количественных данных обеспечивалось выявление групп коррелирующих параметров, восстановление значений параметров систем организма. Необходимость в восстановлении значений определяется нерегулярным характером проведения измерений;

- процессы обработки текстовых медицинских данных. В рамках процессов обработки текстовых данных выполнялись эмпирические процедуры предварительной обработки, основанные на проверке регулярных выражений, а также выявление используемых в текстах терминов. Для текстов, прошедших предварительную обработку, выполнялось восстановление их синтаксической и семантической структуры, осуществлялся поиск характерных текстовых конструкций;

- процессы расчета интегральных показателей состояния пациентов. Выполнялся расчет функционалов на основе множества исходных результатов измерений параметров;

- процессы выявления взаимосвязей в медицинских данных. Между исходными параметрами или отдельными их фрагментами устанавливались статистические, функциональные и логические связи;

- процессы построения представлений медицинских данных, ориентированных на решение одной задачи или группы задач.

*Оценки результатов построения моделей.* В области практической и исследовательской медицины значительно увеличено число решаемых прикладных задач в результате модификации постановок существующих задач, а также рассмотрения принципиально новых, ранее не решавшихся задач, требующих применения методов системного анализа.

Результаты обработки данных о кардиологических больных, собранных более, чем за десять лет, показали, что при решении задач комплексной оценки состояния пациентов в 99% случаев формируемые системой решения точно отражают динамику изменения состояния пациентов. В 1% случаев значения показателей оказались близкими к константным, что свидетельствует о противоречивости исходных данных.

Также применение ППСМ позволило на порядок повысить результативность назначения диагностических процедур. Строились модели, характеризующие состояния пациентов с различными диагнозами, на основе данных ЭХОКГ исследований. Точность моделей составляла от 96 - 98 %. Для построения каждой модели проводилась обработка нескольких тысяч записей.

В результате обработки массивов данных, накопленных в БД существующей МИС, был выявлен целый ряд зависимостей, в том числе новых. Проводился анализ результатов измерений параметров основных систем организма. Существование закономерностей было подтверждено медицинскими специалистами, обладающими экспертными знаниями.

При обработке медицинских текстов за счет использования выявленных шаблонов было обеспечено восстановление синтаксиса и семантики в 94 % записей врачей.

Проведенная систематизация и формализация медицинских данных обеспечила возможность получения и использования связанных данных из открытых источников. Необходимость в дополнительных данных возникает, в частности, при постановке диагнозов, назначении лекарственных препаратов.

#### **5.4.4. Результаты построения моделей объектов по данным мониторинга для области океанографии**

ППСМ в области океанографии ориентированы на обработку результатов измерений параметров состояния Мирового океана и его локальных областей. Потребность в новых моделях, методах и средствах обработки данных в области океанографии определяется

многими изменениями, произошедшими во внешней среде, которые существующие модели не учитывают; значительным увеличением объема собираемых океанографических данных; необходимостью учитывать состояние водной среды при решении прикладных задач в предметных областях.

Методика построения моделей, а также полученные результаты в полном объеме приведены в монографии [341].

Построение моделей было направлено на решение общей задачи предоставления объективных данных о состоянии водной среды для заданных интервала времени, географической области, диапазона глубин.

Задачи оценки состояния водной среды решались в интересах создания новых систем освещения обстановки, построенных на основе интеллектуальных ГИС (ИГИС). Работы велись научно-исследовательской лабораторией СПИИРАН [418] в рамках выполнения серии государственных заказов на протяжении нескольких лет [419]–[422]. Полученные в ходе разработки ИГИС результаты нашли отражение в монографии [312], а также в статьях [194], [203], [205]–[207], [210], [212], [213], [324], [325], [423]–[430].

В рамках общей задачи получения объективных данных о состоянии водной среды проводилось исследование состояния водной среды Арктического региона [431]–[435]. Работы проводились совместно со специалистами ФГБУ «Арктический и антарктический научно-исследовательский институт» (АНИИ) [436] при поддержке международного гранта [437]. Данные для обработки предоставлялись АНИИ. Отдельные работы выполнялись по анализу ледовой обстановки и анализу состояния атмосферы [97].

Отдельно решались задачи обработки данных, получаемых от специализированных средств. Ставилась задача обработки данных от автономных обитаемых подводных аппаратов (АНПА), которые предназначены для обследования морского дна ограниченной акватории [438], [439]. Работы проводились совместно с ДВО РАН [440].

*Состав исходных данных для построения моделей.* Исходными данными являются:

- измерения значений параметров водной среды, выполняемые различными измерителями (океанографическими буями, измерителями, установленными на кораблях и т.д.). Одним из основных источников оперативных данных являлась крупная сеть ныряющих буюв Арго [441];

- данные о состоянии водной среды, собранные в ходе экспедиций;

- обработанные океанографические данные, предоставляемые существующими центрами сбора и обработки данных [442]. Данные могут пройти один или несколько этапов обработки [443]. При их использовании следует учитывать, что оперативная обработка



данных занимает от 12 до 24 часов; учет статистических данных осуществляется только на этапе отложенной обработки;

- данные, получаемые от специализированных средств измерений. При использовании АНПА такими данными являлись данные видеосъемки, гидролокаторов, датчиков, измеряющих расстояния от аппарата до дна и водной поверхности, профилографов, результаты измерений различных параметров водной среды, полученные с использованием установленных на АНПА датчиков;

- базы данных океанографических параметров, содержащие поля данных с фиксированной структурой [444], [445] (гридированные поля данных). При построении полей выполняется осреднение данных на временных интервалах, строится регулярная сеть. Значения параметров рассчитываются для узлов сети. Сеть характеризуется расстоянием между узлами, которое задается в градусах, например, значения могут рассчитываться с шагом  $0.25^\circ$  [446]. Обновление баз данных океанографических параметров требует значительного времени, может проводиться в течение нескольких лет.

*Состав синтезированных моделей.* В области океанографии был выполнен синтез следующих моделей:

- гридированные поля данных по отдельным районам. При построении моделей были выделены области с относительно однородной структурой. Для каждой из областей строилась своя модель, затем модели согласовывались между собой;

- оперативные модели состояния водной среды на основе данных измерителей. Модели строились на основе оперативных данных, поступающих от глобальных сетей океанографических буев. Результирующие модели содержат данные, отвечающие требованиям, предъявляемым к их качеству. При оценке качества выполнялся сравнительный анализ полученных профилей параметров;

- специализированные модели для отдельных районов. Выполнялось построение моделей для отдельных районов по данным, полученным с использованием специализированных средств. Модели включают данные о состоянии водной среды, а также состоянии дна и находящихся на дне объектов;

- базы данных реанализа, содержащие данные по температуре и солёности водной среды. Были построены многоуровневые поля данных с регулярной и нерегулярной структурами. Предусмотрено обновление БД при получении новых данных.

- модели параметров водной среды. Модели отражают зависимости изменения значений параметров при изменении глубины измерений, времени проведения измерений, состояния водной среды других районов, а также состояния водной поверхности и атмосферы.

*Состав синтезированных процессов сбора и обработки океанографических данных.*

Выполнен синтез следующих процессов и обработки океанографических данных:

- процессы оценки качества океанографических данных (процесс верификации данных). Выполнялась оценка качества данных по каждому из источников на основе специализированных наборов тестов, исключались дублирующие значения, проводилась статистическая оценка данных;

- процессы обработки данных. Процессы предусматривали выполнение статистической и интеллектуальной обработки данных отдельных профилей и их групп, а также применение специализированных методов обработки. Специализированная обработка проводилась при работе с отдельными типами измерительных средств. После обработки выполнялась интерполяция данных на стандартные глубины с использованием методов интерполяции [447];

- процессы регуляризации данных. Процесс регуляризации данных включал построение статистических моделей океанографических параметров, построение регулярных и не регулярных полей данных и реанализ. Процесс реанализа строился на основе совместного использования методов множественной оптимальной интерполяции (EOI) [448], трехмерной или четырехмерной вариативной ассимиляции (3DVar/4DVar) [449] и фильтра Калмана (EnK) [450];

- процессы уточнения полей океанографических данных при получении новых данных. Процессы позволяли выполнять системную и локальную перестройки моделей;

- процессы слияния данных, позволяющие получить сводные данные о состоянии водной среды более крупных районов по моделям областей, входящих в их состав;

- процессы исследования динамики изменения состояния параметров водной среды в пространстве и во времени в различных условиях, выявление факторов, влияющих на изменения.

*Оценки результатов построения моделей.* В области океанографии усовершенствованы существующие модели, методы и системы, применяемые при обработке океанографических данных за счет включения в их состав адаптивных процедур, внесения изменений в их конфигурацию, настройки параметров.

По результатам обработки данных по Баренцеву морю были построены гридированные поля данных с разрешением  $0.25 \times 0.25^\circ$ . Обработывались данные по температуре и солености, собранные в период с 1900 г. по 2011 г. При измерениях использовалось 135 800 океанографических станций. Общее число измерений превышает 2 млн. Все измерения были проведены на глубине до 400 м. Применение построенных полей позволило увеличить достоверность предоставляемых данных о состоянии водной среды в

среднем на 15 - 17 %, точность по некоторым районам повысилась на 8 - 15 % и достигла 92%.

Также обеспечено сокращение времени передачи оперативных данных конечным пользователям. Время передачи данных от сети океанографических буев без использования ППСМ составляло от 12 до 24 часов. Внедрение новой системы позволило передавать потребителям порядка 85% данных практически без задержек.

Разработанные модели и методы обеспечили возможность получения оперативных данных о состоянии водной среды в системах освещения обстановки. В таких системах оперативные данные необходимы для решения ряда прикладных задач, включая расчет и выдачу информации для обеспечения безопасности плавания и маневрирования кораблей и судов, расчет зон обнаружения целей различных классов РЭС кораблей в условиях изменяющейся обстановки и другие. Кроме того, наличие нового источника данных позволило расширить состав данных, отображаемых пользователям при освещении обстановки.

Исследование океанографических данных с применением предложенных моделей и методов позволило построить динамические модели водной среды для отдельных районов. Построенные модели позволили выявить отдельные закономерности в состоянии водной среды. Возможность существования выявленных закономерностей подтверждена предметными специалистами, обладающими экспертными знаниями.

#### **5.4.5. Оценка результатов построения моделей объектов по данным мониторинга и возможности дальнейшего использования**

Анализ результатов построения моделей объектов космического назначения, телекоммуникаций, практической и исследовательской медицины и океанографии по данным мониторинга показал, что предложенные модели и методы позволяют получить существенный выигрыш по сравнению с известными решениями (таблица 5.3).

По оценкам привлекаемых предметных специалистов, обладающих экспертными знаниями, достигаемые показатели (таблица 5.3) во многом удовлетворяют современным потребностям практики.

Результаты применения многоуровневого синтеза моделей объектов мониторинга при решении практических задач в рассмотренных предметных областях, а также при решении ряда других задач, приведены в отчетных материалах по НИР и ОКР [347]–[362], [419]–[422], [437], [451]–[459]. Задача решались с применением разработанных программных комплексов и отдельных программ [326, 483].

Значительный интерес к предлагаемым моделям и методам многоуровневого синтеза проявляют специалисты в следующих областях: области городского хозяйства [461]–[468], области образования [469]–[474], области психологии [475].

Таблица 5.3 – Оценки показателей эффективности решения прикладных задач в предметных областях

Показатель эффективности решения прикладных задач	Прикладная область			
	Объекты космического назначения	Океанография	Практическая и исследовательская медицина	Телекоммуникации
Точность результатов	98%	увеличена на 8-15%.	99%	-
Достоверность результатов	85%-95%	увеличена на 15-17%	-	-
Затрачиваемое время	сокращено до секунд и минут (ранее часы / дни)	сокращено до минут (ранее часы / дни)	секунды (ранее часы)	-
Стоимость проектирования, разработки, сопровождения систем	снижена в два и более раз	-	-	снижена в два и более раз
Число успешно решаемых прикладных задач	-	-	98%	98%

Некоторые частные задачи мониторинга с применением новых методов были решены в банковской сфере [476].

Предложенные модели и методы обработки данных оказались востребованы при обработке геофизических данных специалистами института вулканической геологии и геохимии ДВО РАН [477]–[480]. Отдельные идеи использовались при разработке алгоритмов обработки речи [481].

Элементы разработанных архитектурных решений применялись при создании новых информационных систем в области культуры [482].

## 5.5. Выводы

1. Разработана новая система методик построения моделей объектов по данным мониторинга, включающая общую методику и частные методики для предметных областей. В частных методиках учитываются состав решаемых задач, условия построения моделей, состав обрабатываемых данных, а также традиционные способы их обработки. Для предметных областей разработаны новые прикладные модели, методы и алгоритмы построения моделей, позволяющие выполнять обработку данных предметных областей при

синтезе моделей объектов. Процессы построения моделей поддержаны инструментально. Разработано программное средство, которое позволяет адаптировать ППСМ для работы с ним специалистов предметных областей.

2. Сформулированы рекомендации по построению моделей объектов мониторинга. Предложен скриптовый язык описания синтезируемых программ мониторинга. Построены онтологии обработки данных, включающие существующие и вновь разработанные модели и методы обработки данных мониторинга. Разработана открытая программная платформа для построения моделей объектов.

3. Разработаны системы построения моделей объектов мониторинга для четырех предметных областей: области объектов космического назначения, области океанографии, области практической и исследовательской медицины и области телекоммуникаций. Системы разрабатывались в рамках ОКР и коммерческих проектов. Значительная часть систем находится в опытной эксплуатации. Период их эксплуатации составляет более трех лет. Все системы востребованы на объектах эксплуатации.

4. Получены количественные и качественные оценки эффективности ППСМ с привлечением специалистов предметных областей, обладающих экспертными знаниями.

В области объектов космического назначения обеспечено повышение точности и достоверности результатов обработки данных, получаемых от наблюдаемых объектов, существенно сокращено время их обработки. Оценки даны специалистами космодромов и полигонов РФ. Результаты обработки данных РКН семейства СОЮЗ показали, что ППСМ обеспечивают достоверное решение 85%-95% прикладных задач, точность обработки составляет порядка 98%, время обработки, как правило, занимает секунды, не превышает нескольких минут.

В области океанографии усовершенствованы существующие модели, методы и системы, применяемые при мониторинге за счет включения в их состав адаптивных процедур, внесения изменений в их конфигурацию, настройки параметров. Оценки даны специалистами Арктического и антарктического научно-исследовательского института (АНИИ). Результаты обработки данных по Баренцеву морю показали, что достоверность получаемых решений была увеличена в среднем на 15-17%, точность повысилась на 8-15%. Также обеспечено сокращение времени передачи оперативных данных конечным пользователям. Время передачи данных от сети океанографических буев без использования ППСМ составляло от 12 до 24 часов. Внедрение новой системы позволило передавать потребителям порядка 85% данных практически без задержек.

В области практической и исследовательской медицины значительно увеличено число решаемых прикладных задач в результате модификации постановок существующих задач, а

также рассмотрения принципиально новых, ранее не решавшихся задач, требующих использования новых прикладных теорий. Оценки даны специалистами ФГБУ «НМИЦ им. В. А. Алмазова». Результаты обработки данных кардиологических больных, собранных более, чем за десять лет, показало, что при решении задач комплексной оценки состояния пациентов в 99% случаев формируемые системой решения точно отражают динамику изменения состояния пациентов. В 1% случаев значения показателей оказались близкими к константным, что свидетельствует о противоречивости исходных данных. Также ППСМ позволили на порядок повысить результативность назначения диагностических процедур. Для эхокардиографического исследования этот показатель достиг 98%.

В области телекоммуникаций существенно расширена область применения систем мониторинга за счет низкой стоимости их проектирования и сопровождения, а также удобства работы с ними конечных пользователей. Оценки эффективности систем даны на основе обработки статистических данных об эксплуатации систем мониторинга. Системы были развернуты в нескольких крупных и средних сегментах сетей операторов кабельного телевидения Северной Америки и Канады. Общее число контролируемых пользовательских устройств составляло несколько миллионов. Результаты обработки данных об эксплуатации систем показали, что системы обеспечили своевременное выявление и устранение 98,5% всех ошибок.

В целом, в пятой главе разработана новая система методик построения моделей объектов по данным мониторинга для решения прикладных задач.

## Заключение

Решена научная проблема разработки основ теории и методов многоуровневого автоматического синтеза автоматных моделей объектов мониторинга, отвечающих современным потребностям практики по снижению вычислительной сложности этого синтеза, имеющая существенное значение для развития народно-хозяйственной и оборонной отраслей. Получены следующие научные результаты:

1. Основы новой теории многоуровневого автоматического синтеза автоматных моделей объектов мониторинга, отличающейся оригинальными: концептуальной моделью синтеза многоуровневых перестраиваемых автоматных моделей, системой показателей и критериев эффективности, предусматривающей оценку полноты моделей и сложности их построения, формализмами математического описания иерархических относительно-конечных автоматов, математическими формулировками задач многоуровневого синтеза.

2. Новые индуктивно - дедуктивные методы многоуровневого автоматического синтеза автоматных моделей объектов мониторинга, основанные на новых постановках задач и методах их решения, включающих метод многоуровневого индуктивного синтеза, позволяющего строить модели объектов по поступающим от них данным, и метод многоуровневого дедуктивного синтеза, позволяющий доказывать существование новых, ранее не рассматривавшихся моделей, в пространстве, построенном в результате индуктивного синтеза.

3. Новые методы многоуровневого автоматического синтеза автоматных моделей процессов и программ мониторинга, отличающиеся низкой вычислительной сложностью, обеспеченной за счет разработки оригинальных методов, позволяющих строить процессы мониторинга, основанные на доказательстве их существования с применением прямого нисходящего многоуровневого вывода, и программы мониторинга на основе обратного многоуровневого вывода.

4. Новые методы и модели многоуровневой трансформации данных, обеспечивающие возможность контентно адаптивной обработки результатов мониторинга за счет описания процессов обработки в общем виде и их поэтапной детализации до уровня программно-реализуемых с учетом содержания данных и условий синтеза моделей объектов.

5. Оригинальные методы и модели разработки проблемно- и предметно-ориентированных систем, описываемых в виде иерархии согласованных архитектурных и онтологических моделей, что позволяет применять гибкие методологии при их проектировании, создании и сопровождении.

6. Новая система методик построения моделей объектов по данным мониторинга для решения прикладных задач, отличающаяся от существующих тем, что обеспечивает полноту

синтезируемых прикладных моделей и низкую сложность их синтеза, позволяет решать практические задачи с учетом их классов, показателей и критериев эффективности, используемых для оценки формируемых результатов, а также типов данных, обрабатываемых в предметных областях.

Результаты апробации разработанного аппарата многоуровневого синтеза при построении моделей объектов космического назначения, телекоммуникаций, практической и исследовательской медицины, океанографии с привлечением специалистов предметных областей подтвердили, что предложенные решения позволяют на практике обеспечить достижение требуемых показателей эффективности, включающих число успешно решенных задач, точность и достоверность решений, время, затрачиваемое на их формирование, а также стоимость проектирования систем.

**Результаты диссертационного исследования соответствуют следующим пунктам паспорта специальности 05.13.01 (технические системы):**

1. Теоретические основы и методы системного анализа, оптимизации, управления, принятия решений и обработки информации.

2. Формализация и постановка задач системного анализа, оптимизации, управления, принятия решений и обработки информации.

4. Разработка методов и алгоритмов решения задач системного анализа, оптимизации, управления, принятия решений и обработки информации.

7. Методы и алгоритмы структурно-параметрического синтеза и идентификации сложных систем.

9. Методы разработка проблемно-ориентированных систем управления, принятия решений и оптимизации технических объектов.

**Рекомендации, перспективы дальнейшей разработки темы диссертационных исследований.**

Результаты диссертационного исследования могут быть использованы в перспективных НИР и ОКР, а также в учебном процессе при подготовке специалистов, связанных с проблемами автоматизации мониторинга различных объектов и построения их моделей. Они также могут быть востребованы при модернизации существующих систем мониторинга.

Дальнейшие исследования по тематике диссертации целесообразно проводить в направлении развития многоуровневого автоматического синтеза процессов мониторинга, ориентированных на решение прикладных задач в предметных областях.



## Список сокращений и условных обозначений

БП – бизнес процесс  
БС - бизнес сервисы  
ВС – внешняя среда  
ГТС - групповой телеметрический сигнал  
ИГИС – интеллектуальные информационные системы  
И - результаты измерений  
ИИ - искусственный интеллект  
ИС - информационная система  
ИУС - информационно-управляющая система  
ОМ – объект мониторинга  
М - модель  
МАС - мультиагентная система  
ММС – машина многоуровневого синтеза  
МО – массовое обслуживание  
НГО - нагруженный граф отношений  
ОиА – область обработки и анализа данных  
ОМ – объект мониторинга  
ОКА – относительно конечный операционный автомат  
ППР – поддержка принятия решений  
ППСМ – проблемно- и предметно- ориентированная система построения моделей объектов  
СОА – сервисно-ориентированная архитектура  
СОАИ - система обработки и анализа результатов измерений  
ТМИ - телеметрическая информация  
ЦС - целевая система  
DF - Data Fusion  
DM – Data Mining  
DSL - Domain Specific Language  
MDA - Model Driven Architecture  
MDE - Model Driven Engineering  
ML – Machine Learning  
MM –Metamodel  
MT - Model transformation  
MW - Model weaving  
OMP - Open Modelling Platform  
SW - SemanticWeb

**СПИСОК ЛИТЕРАТУРЫ**

1. Когнитивные информационные системы мониторинга [Текст]: Монография / А. В. Васильев, А. И. Вайнтрауб, А. И. Водяхо, Н. А. Жукова, Д. И. Курапеев, М. С. Лушнов, Н. А. Смирнов. – СПб.: Изд-во СПбГЭТУ «ЛЭТИ», 2017. – 200 с.
2. Когнитивный мониторинг телекоммуникационных сетей [Текст]: Монография / В. Ю. Осипов, А. Калмацкий, А. И. Водяхо, Н. А. Жукова, П. А. Глебовский. – СПб.: Изд-во СПбГЭТУ «ЛЭТИ», 2018. – 202 с.
3. EMC's reports [Электронный ресурс]. – Режим доступа: <https://www.emc.com/>
4. Gartner's reports [Электронный ресурс]. – Режим доступа: <http://www.gartner.com/>
5. Linked Data [Электронный ресурс]. – 2018. – Режим доступа: <http://www.linkeddata.org>
6. Horizon 2020. Information and Communication Technologies [Электронный ресурс]. – Режим доступа: <https://ec.europa.eu/programmes/horizon2020>
7. IRIG 106 [Электронный ресурс]. – Режим доступа: <http://www.irig106.org/>
8. World Wide Web Consortium (W3C) [Электронный ресурс]. – Режим доступа: <https://www.w3.org/>
9. Жукова, Н. А. Контексты адаптивной обработки многомерных измерений параметров пространственно соотнесенных объектов / Н. А. Жукова // III Междунар. науч.-техн. и науч.-метод. конф. «Актуальные проблемы инфотелекоммуникаций в науке и образовании» (СПб., 25–26 февр. 2014 г.): Сб. ст. – СПб.: Изд-во СПбГУТ, 2014. – С. 465–471.
10. Жукова, Н. А. Проблема когнитивного мониторинга распределенных объектов / Н. А. Жукова, Н. Р. Андриянова // Научно-техническая информация. Сер. 2, Информационные процессы и системы. – 2019. – № 2. – С. 18–29.
11. Zhukova, N. A. Cognitive Monitoring of Distributed Objects / N. A. Zhukova, N. R. Andriyanova // Automatic Documentation and Mathematical Linguistics. – 2019. – Vol. 53, Iss. 1. – P. 32–43.
12. Охтилев, М. Ю. Интеллектуальные технологии мониторинга и управления структурной динамикой сложных технических объектов [Текст] / М. Ю. Охтилев, Б. В. Соколов, Р. М. Юсупов. – М.: Наука, 2006. – 408 с.
13. Микони, С. В. Квалиметрия моделей и полимодельных комплексов [Текст]: Монография / С. В. Микони, Б. В. Соколов, Р. М. Юсупов. – М.: Наука, 2018.
14. Kotseruba, I. 40 years of cognitive architectures: core cognitive abilities and practical applications / I. Kotseruba, J. K. Tsotsos // Artificial Intelligence Review. – 2018. – Vol. 52. – P. 1–78.

15. Hawkins, J. On Intelligence. Broun Walker / J. Hawkins, S. Blakeslee. – 2006 [Электронный ресурс]. – Режим доступа: [https://www.goodreads.com/book/show/27539.On\\_Intelligence](https://www.goodreads.com/book/show/27539.On_Intelligence)
16. Arel, I. Destin: a scalable deep learning architecture with application to high-dimensional robust pattern recognition / I. Arel, D. Rose, R. Coop // Proceedings of the AAAI Workshop on Biologically Inspired Cognitive Architectures. – Nov. 2009 [Электронный ресурс]. – Режим доступа: <http://web.eecs.utk.edu/~itamar/Papers/BICA2009.pdf>
17. O'Reilly, R. C. A biologically-based computational model of working memory / R. C. O'Reilly, T. S. Braver, J. D. Cohen; Ed. by A. Miyake, P. Shah // Models of Working Memory. – 1999. – P. 375–411 [Электронный ресурс]. – Режим доступа: <https://www.cambridge.org/core/books/models-of-working-memory/biologically-based-computational-model-of-working-memory/9075422601EF415B329D4BB1600F8DAE>
18. Retrospective and prospective responses arising in modeled hippocampus during maze navigation by a brain-based device / J. Fleischer, J. Gally, G. Eldeman, J. Krichmar // Proceedings of the National Academy of Sciences. – 2007. – Vol. 104 [Электронный ресурс]. – Режим доступа: <https://www.semanticscholar.org/paper/Retrospective-and-prospective-responses-arising-in-Fleischer-Gally/f81e89f89b207723efacdbf92dddb3e0425b147d>
19. Modha, S. Introducing a Brain-inspired Computer TrueNorth's neurons to revolutionize system architecture / S. Modha. – IBM, 2014 [Электронный ресурс]. – Режим доступа: <http://www.research.ibm.com/articles/brain-chip.shtml>
20. Cognitive Control / S. Haykin, M. Fatemi, P. Setoodeh, Y. Xue // Proceedings of the IEEE. – 2012. – Vol. 100. – P. 3156–3169 [Электронный ресурс]. – Режим доступа: <https://doi.org/10.1109/JPROC.2012.2215773>
21. Haikonen, P. The Role of Associative Processing in Cognitive Computing / P. Haikonen // Cognitive Computation. – 2009. Vol. 1. – P. 42–49 [Электронный ресурс]. – Режим доступа: <https://doi.org/10.1007/s12559-009-9006-y>
22. Palm, G. Neural associative memories and sparse coding / G. Palm // Neural Networks. – 2013. – Vol. 37. – P. 165–171 [Электронный ресурс]. – Режим доступа: <http://dx.doi.org/10.1016/j.neunet.2012.08.013>
23. Haykin, S. Neural Networks and Learning Machines / S. Haykin. – 3rd ed. – New York: PrenticeHall, 2008.
24. A biologically inspired spiking model of visual processing for image feature detection / D. Kerr, T. M. McGinnity, S. Coleman, M. Clogenson // Neurocomputing. – 2015. – Vol. 158. P. 268–280. – ISSN 0925-2312 [Электронный ресурс]. – Режим доступа: <https://doi.org/10.1016/j.neucom.2015.01.011>

25. A historical survey of algorithms and hardware architectures for neural-inspired and neuromorphic computing applications / C. D. James, J. B. Aimone, N. E. Miner, C. M. Vineyard, F. H. Rothganger, K. D. Carlson, S. A. Mulder, T. J. Draelos, A. Faust, M. J. Marinella, J. H. Naegle, S. J. Plimpton // *Biologically Inspired Cognitive Architectures*. – 2017. – Vol. 19. P. 49–64. – ISSN 2212-683X [Электронный ресурс]. – Режим доступа: <https://doi.org/10.1016/j.bica.2016.11.002>

26. A world survey of artificial brain projects, Part II: Biologically inspired cognitive architectures / B. Goertzel, R. Lian, I. Arel, H. de Garis, S. Chen // *Neurocomputing*. – 2010. – Vol. 74, Iss. 1–3. – P. 30–49. – ISSN 0925-2312 [Электронный ресурс]. – Режим доступа: <https://doi.org/10.1016/j.neucom.2010.08.012>

27. Treisman, A. The binding problem / A. Treisman // *Current Opinion in Neurobiology*. – 1996. – Vol. 6. P. 171–178.

28. Osipov, V. Space-time signal binding in recurrent neural networks with controlled elements / V. Osipov, M. Osipova // *Neurocomputing*. – 2018. Vol. 308. P. 194–204 [Электронный ресурс]. – Режим доступа: <https://doi.org/10.1016/j.neucom.2018.05.009>

29. Osipov, V. Formal aspects of streaming recurrent neural networks / V. Osipov, V. Nikiforov; Ed. by T. Huang et al. // *Advances in Neural Networks, LNCS*. – 2018. – Vol. 10878. P. 29–36 [Электронный ресурс]. – Режим доступа: <https://www.springer.com/us/book/9783319925363>

30. Osipov, V. Space-time structure of recurrent neural networks with controlled synapses / V. Osipov; Ed. by L. Cheng, et al. // *Advances in Neural Networks, LNCS*. – 2016. – Vol. 9719. P. 177–184 [Электронный ресурс]. – Режим доступа: [https://link.springer.com/chapter/10.1007/978-3-319-40663-3\\_21](https://link.springer.com/chapter/10.1007/978-3-319-40663-3_21)

31. Osipov, V. Structure and basic functions of cognitive neural network machine / V. Osipov // *MATEC Web of Conferences*. – 20 June 2017. – Vol. 113 [Электронный ресурс]. – Режим доступа: <https://doi.org/10.1051/mateconf/201711302011>

32. Осипов, В. Ю. Ассоциативная интеллектуальная машина с тремя сигнальными системами / В. Ю. Осипов // *Информационно-управляющие системы*. – 2014. – № 5 (72). С. 12–17 [Электронный ресурс]. – Режим доступа: <https://cyberleninka.ru/article/n/assotsiativnaya-intellektualnaya-mashina-s-tremya-signalnymi-sistemami>

33. Способ интеллектуальной обработки информации в нейронной сети [Текст] / В. Ю. Осипов и др. Пат. RU 2553074, опубл. 10.06.2015; Пат. RU 2514931, опубл. 10.05.2014; Пат. RU 2483356, опубл. 27.05.2013; Пат. RU 2502133, опубл. 20.12.2013; Пат. RU 2446463, опубл. 27.03.2012; Пат. RU 2413304, опубл. 27.02.2011; Пат. RU 2427914, опубл. 27.08.2011.

34. An Integrated Theory of the Mind / J. R. Anderson, D. Bothell, M. D. Byrne, S. Douglass, C. Lebiere, Y. Qin // *Psychological Review*. – Oct. 2004. – Vol. 111 (4). – P. 1036–1060.
35. Laird, J. E. *The Soar Cognitive Architecture* / J. E. Laird. – Cambridge, Massachusetts: MIT Press, 2012.
36. Fusion paradigms in cognitive technical systems for human–computer interaction / M. Glodek, F. Honold, T. Geier, G. Krell, F. Nothdurft, S. Reuter, F. Schüssel, T. Hörnle, K. Dietmayer, W. Minker, S. Biundo, M. Weber, G. Palm, F. Schwenker // *Neurocomputing*. – 2015. – Vol. 161. P. 17–37. – ISSN 0925-12 [Электронный ресурс]. – Режим доступа: <https://doi.org/10.1016/j.neucom.2015.01.076>
37. Perlovsky, L. I. Cognitive high level information fusion / L. I. Perlovsky // *Information Sciences*. – 2007. – Vol. 177. – P. 2099–2118.
38. Land, G. *Grow or Die: The Unifying Principle of Transformation* / G. Land. –Leadership 2000 Inc., 1997.
39. A novel bit level time series representation with implication of similarity search and clustering / Ch. Ratanamahatana, et al. // *Advances in Knowledge Discovery and Data Mining*. (May 18–20 2005, Hanoi, Vietnam). 9th Pacific-Asia Conference on Knowledge Discovery and Data Mining. – Berlin, Heidelberg: Springer, 2005. – P. 771–777.
40. Chan, Kin-Pong. Efficient time series matching by wavelets / Kin-Pong Chan, Fu Wai-Chee // *Proceedings 15th International Conference on Data Engineering* (Sydney, Australia, March 23–26, 1999). – IEEE, 1999. – C. 126–133.
41. Dimensionality reduction for fast similarity search in large time series databases / E. Keogh, et al. // *Knowledge and information Systems*. – 2001. – Vol. 3.3. – P. 263–286.
42. Experiencing SAX: a novel symbolic representation of time series / J. Lin, et al. // *Data Mining and knowledge discovery*. 2007. – Vol. 15.2. – P. 107–144.
43. Kawahara, Y. Change-Point Detection in Time-Series Data by Direct Density-Ratio Estimation / Y. Kawahara, M. Sugiyama // *Society for Industrial and Applied Mathematics – 9th SIAM International Conference on Data Mining 2009, SDM* (April 30 – May 2 2009, Sparks, USA). – 2009. – Vol. 1. – P. 385–396.
44. Turner, R. Adaptive Sequential Bayesian Change Point Detection / R. Turner, Y. Saatei, C. E. Rasmussen // *Temporal Segmentation Workshop at NIPS*. – Dec. 2009. – P. 1–4.
45. Ye, L. Time series shapelets: a new primitive for data mining / L. Ye, E. Keogh // *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining* (June 28 – July 01 2009, Paris, France). – ACM, 2009. – C. 947–956.

46. Ye, L. Time series shapelets: a novel technique that allows accurate, interpretable and fast classification / L. Ye, E. Keogh // *Data mining and knowledge discovery*. – 2011. – Vol. 22.1–2. – P. 149–182.
47. Azzouzi, M. Analysing time series structure with Hidden Markov Models / M. Azzouzi, T. N. Ian // *Neural Networks for Signal Processing VIII. Proceedings of the IEEE Signal Processing Society Workshop (Cat. No. 98TH8378) (Sept. 2, 1998)*. – Cambridge, UK, 1998. – P. 402–408.
48. Nanopoulos, A. Feature-based classification of time-series data / A. Nanopoulos, R. Alcock, Y. Manolopoulos // *International Journal of Computer Research*. – 2001. – Vol. 10.3. – P. 49–61.
49. Kalpakis, K. Distance measures for effective clustering of ARIMA time-series. / K. Kalpakis, D. Gada, V. Puttagunta // *Data Mining, 2001. ICDM 2001, Proceedings IEEE International Conference on Data Mining (San Jose, USA, 29 Nov. – 2 Dec., 2001)*. – IEEE, 2001. – C. 273–280.
50. Susto, G. A. Time-series classification methods: Review and applications to power systems data / G. A. Susto, A. Cenedese, M. Terzi // *Big Data Application in Power Systems Elsevier*. – Elsevier Inc., 2018. – P. 179–220.
51. Sorzano, C. A survey of dimensionality reduction techniques / C. Sorzano, J. Vargas, A. Pascual-Montano. – 2014 [Электронный ресурс]. – Режим доступа: <https://arxiv.org/abs/1403.2877>
52. Бокс, Дж. Анализ временных рядов прогноз и управление [Текст] / Дж. Бокс, Г. Дженкинс: Пер. с англ. // Под ред. В.Ф. Писаренко. – Т.1–2. – М.: Мир, 1974.
53. Russell, S. *Artificial Intelligence: A Modern Approach* / S. Russell, P. Norvig. – 3 rd ed. – Pearson, 2009.
54. Искусственный интеллект [Текст]: в 3 кн. Кн. 2. Модели и методы: Справ. / Под ред. Д. А. Поспелова. – М.: Наука, 1990. – 304 с.
55. Hastie, T. *The Elements of Statistical Learning: Data Mining, Inference, and Prediction* / T. Hastie, R. Tibshirani, J. Friedman. – 2nd ed. – Springer-Verlag, 2009. – P. 746.
56. Witten, I. H. *Data Mining: Practical Machine Learning Tools and Techniques* / I. H. Witten, E. Frank. – 2nd ed. – Morgan Kaufmann, 2016.
57. *Data Mining and Analysis: Fundamental Concepts and Algorithms* / M. Zaki, et al. – Cambridge University Press, 2014.
58. Graesser, L. *Learning Machine Learning* / L. Graesser. – Retrieved. Nov. 3, 2018.
59. Extreme learning machine: algorithm, theory and applications / S. Ding, H. Zhao, Y. Zhang, X. Xu, R. Nie // *Artificial Intelligence Review*. – 2015. – Vol. 44 (1). – P. 103–115.

60. Data cleaning: Overview and emerging challenges. / X. Chu, I. F. Piyas, S. Krishnan, J. Wang // Proceedings of the 2016 International Conference on Management of Data. ACM (2016, June). – P. 2201–2206.
61. Городецкий, В. И. Ассоциативная классификация: аналитический обзор. Ч. 1 / В. И. Городецкий, О. Н. Тушканова // Труды СПИИРАН. – 2015. – Вып. 38. – С. 183–203.
62. Городецкий, В. И. Ассоциативная классификация: аналитический обзор. Ч. 2 / В. И. Городецкий, О. Н. Тушканова // Труды СПИИРАН. – 2015. – Вып. 39. – С. 212–240.
63. Formal Concept Analysis: Foundations and Applications / Ed. by V. Ganter, G. Stumme, R. Wille. Vol. 3626. – Berlin, Heidelberg: Springer Berlin Heidelberg, 2005 [Электронный ресурс]. – Режим доступа: <https://doi.org/10.1007/978-3-540-31881-1>
64. Ganter, V. Formal Concept Analysis: Mathematical Foundations / V. Ganter, R. Wille. – Berlin, Heidelberg: Springer Berlin Heidelberg, 1999 [Электронный ресурс]. – Режим доступа: <https://doi.org/10.1007/978-3-642-59830-2>
65. Загоруйко, Н. Г. Когнитивный анализ данных / Н. Г. Загоруйко. – Новосибирск: Академ. изд-во «ГЕО», 2012. – 203 с.
66. Загоруйко, Н. Г. Прикладные методы анализа данных и знаний / Н. Г. Загоруйко. – Новосибирск: ИМ СО РАН, 1999. – 270 с.
67. Загоруйко, Н. Г. Алгоритмы обнаружения эмпирических закономерностей / Н. Г. Загоруйко, В. Н. Елкина, Г. С. Лбов. – Новосибирск: Наука, 1985.
68. Загоруйко, Н. Г. Методы распознавания и их применение / Н. Г. Загоруйко. – М.: Сов. радио, 1972.
69. Povinelli, R. Time series data mining: Identifying temporal patterns for characterization and prediction of time series events / R. Povinelli // TSDM'00 Proceedings of the First International Workshop on Temporal, Spatial, and Spatio-Temporal Data Mining-Revised Papers. – London: Springer-Verlag, 2000. – P. 46–61.
70. Scargle, J. Bayesian Blocks for Particle Physics / J. Scargle [Электронный ресурс]. – Режим доступа: [http://people.stat.sfu.ca/~lockhart/richard/banff2010/scargle\\_bb.pdf](http://people.stat.sfu.ca/~lockhart/richard/banff2010/scargle_bb.pdf)
71. Dobigeon, N. Joint Segmentation of Multivariate Astronomical Time Series: Bayesian Sampling With a Hierarchical Model / N. Dobigeon, J. Tourneret, J. Scargle // IEEE Transactions on Signal Processing. – 2007. – Vol. 55, Iss. 2. – С. 414–423.
72. Extracting interpretable features for early classification on time series / Z. Xing, et al. // Proceedings of the 2011 SIAM International Conference on Data Mining. Society for Industrial and Applied Mathematics, 2011 (April 28–30 2011, Mesa, Arizona, USA). – SIAM, 2011. – Vol. 11. – P. 247–258.

73. Miloud-Aouidate, A. Survey of Nearest Neighbor Condensing Techniques / A. Miloud-Aouidate, A. Algeriababa-Ali // *International Journal of Advanced Computer Science and Applications*. – 2011. – Vol. 2, № 11. – P. 59–64.
74. Tuncel, K. S. Autoregressive forests for multivariate time series modeling / K. S. Tuncel, M. G. Baydogan // *Pattern Recognition*. – 2018. – Vol. 73. – P. 202–215.
75. Allen, J. F. Moments and Points in an Interval-Based Temporal Logic / J. F. Allen, P. J. Hayes // *Computational Intelligence*. – 1989. – Vol. 5 (4). – P. 225–238.
76. Allen, J. F. Actions and Events in Interval Temporal Logic / J. F. Allen, G. Ferguson // *Journal of Logic and Computation*. – 1994. – Vol. 4, Iss. 5. – P. 531–579.
77. Gomes, E. F. Classifying heart sounds using sax motifs, random forests and text mining techniques / E. F. Gomes, A. M. Jorge, P. J. Azevedo // *Proceedings of the 18th International Database Engineering & Applications, 2001* [Электронный ресурс]. – Режим доступа: <http://repositorium.sdum.uminho.pt/bitstream/1822/33769/1/2001.pdf>
78. Whitehouse, K. Semantic streams: A framework for composable inference over sensor data / K. Whitehouse, J. Liu, F. Zhao // *European Workshop on Wireless Sensor Networks (EWSN'06)*. Vol. 3868 of *Lecture Notes in Computer Science* / Ed. by K. Römer, H. Karl, F. Mattern. – 3rd ed. – Zurich, Switzerland: Springer, 2006. – P. 5–20.
79. *Handbook of multisensor data fusion: theory and practice* / Ed. by M. Liggins, D. Hall, J. Llinas. – 2nd ed. – FL. Boca Raton: Taylor & Francis Group, 2009. – P. 849.
80. Friedlander, D. S. Semantic information fusion for coordinated signal processing in mobile sensor networks / D. S. Friedlander, S. Phoha // *International Journal of High Performance Computing Applications*. – 2002. – Vol. 16 (3). – P. 235–241.
81. Blasch, E. *High-Level Information Fusion Management and System Design* / E. Blasch, E. Bosse, D. Lambert. – Artech House Publishers, Norwood, MA, 2012.
82. Wald, L. Some terms of reference in data fusion / L. Wald // *IEEE Transactions on Geosciences and Remote Sensing*. – 1999. – Vol. 37.3. – P. 1190–1193.
83. Raol, J. R. *Multi-Sensor Data Fusion with MATLAB* / J. R. Raol. – Taylor & Francis Group, Boca Raton, FL, 2010.
84. Houzelle, S. Contribution to multisensor fusion formalization / S. Houzelle, G. Girodon // *Robotics and Autonomous Systems*. – 1984. – Vol. 13. – P. 69–85.
85. Das, S. *High-Level Data Fusion* / S. Das. – Norwood, MA: Artech House Inc., 2008. – 373 p.
86. Bedworth, M. D. The omnibus model: A new model for data fusion? / M. D. Bedworth, J. C. O'Brien // *Proceedings of the 2nd International Conference on Information Fusion (FUSION'99)*. – Sunnyvale, USA. ISIF, 1999. – P. 437–444.



87. Rasmussen, J. *Information Processing and Human Machine Interaction: An Approach to Cognitive Engineering* / J. Rasmussen. – New York: North Holland, 1986.
88. Dasarathy, B. V. *Decision Fusion* / B. V. Dasarathy. – Los Alamitos, CA: IEEE Computer Society Pr., 1994.
89. Dasarathy, B. V. *Sensor fusion potential exploitation-innovative architectures and illustrative applications* / B. V. Dasarathy // *Proceedings of the IEEE*. – 1997. – Vol. 85, № 1. – P. 24–38.
90. Bosse, E. *Concepts, Models, and Tools for Information Fusion* / E. Bosse, J. Roy, S. Wark. – Norwood: MA ARTECH House, INC., 2007.
91. Shahbazian, E. *Data Fusion for Situation Monitoring, Incident Detection, Alert and Response Management* / E. Shahbazian, G. Rogova, P. Valin. – Amsterdam: IOS Press, 2005.
92. Castanedo F. *A Review of Data Fusion Techniques* / F. Castanedo // *The Scientific World Journal Volume*. – 2013. – Article ID 704504. – P. 19.
93. *Revisiting the JDL data fusion model II* / C. Llinas, C. Bowman, G. Rogova, A. Steinberg, E. Waltz, F. White. – Technical Report, DTIC Document, 2004.
94. Blasch, E. P. *JDL level 5 fusion model “user refinement” issues and applications in group tracking* / E. P. Blasch, S. Plano // *Proceedings of the Signal Processing, Sensor Fusion, and Target Recognition XI*. – April 2002. – P. 270–279.
95. Steinberg, A. *Revisions to the JDL Data Fusion Model. Sensor Fusion: Architectures, Algorithms, and Applications* / A. Steinberg, C. Bowman, F. White // *Proceedings of the SPIE*. – 1999. – Vol. 3719. – C. 430–441.
96. Karakowski, J. A. *Towards Visual Data Fusion* / J. A. Karakowski // *NSSDF International Open Session*. – Atlanta, GA: Georgia Tech Research Institute, April 1998.
97. Водяхо, А. И. *Адаптивные алгоритмы выявления чрезвычайных ситуаций в условиях ледовой обстановки* / А. И. Водяхо, Н. А. Жукова // *Известия СПбГЭТУ «ЛЭТИ»*. Сер. Информатика, управление и компьютерные технологии. – 2014. – Вып. 10. – С. 15–21.
98. Водяхо, А. И. *Интеллектуальные системы для конфигурирования линеек программных продуктов* / А. И. Водяхо, Н. А. Жукова // *Науч. сессия НИЯУ МИФИ-2015*. – М.: МИФИ, 2015. – Т. 3. – С. 123.
99. Zhukova, N. *Agile Configuration Management: Concept and Techniques* / N. Zhukova // *12th International Conference on Knowledge, Economy and Management – ICKEM2014 (Antalya, Turkey, Nov. 27–30, 2014)*. – Istanbul Medeniyet University and University of Wisconsin Milwaukee. – P. 209–218.
100. *Multilevel network analysis for the social sciences: Theory, methods and applications* / Ed. by E. Lazega, T. A. Snijders. – Vol. 12. – Springer, 2015.

101. Festl, R. The individual or the group: A multilevel analysis of cyberbullying in school classes / R. Festl, M. Scharkow, T. Quandt // *Human Communication Research*. – 2015. – Vol. 41 (4). – P. 535–556.
102. Ayodele, T. O. Types of machine learning algorithms / T. O. Ayodele // *New advances in machine learning*. – InTech, 2010.
103. Panov, P. Ontology of core data mining entities / P. Panov, L. Soldatova, S. Džeroski // *Data Mining and Knowledge Discovery*. – 2014. – Vol. 28.5–6. – P. 1222–1265.
104. Luo, G. A review of automatic selection methods for machine learning algorithms and hyper-parameter values / G. Luo // *Network Modeling Analysis in Health Informatics and Bioinformatics*. – 2016. – Vol. 5.1. – P. 18.
105. A taxonomy of dirty data / W. Kim, B. J. Choi, E. K. Hong, S. K. Kim, D. Lee // *Data mining and knowledge discovery*. – 2003. – Vol. 7 (1). – P. 81–98.
106. Xu, Rui. Survey of clustering algorithms / Rui Xu, D. Wunsch // *IEEE Transactions on neural networks*. – 2005. – Vol. 16.3. – P. 645–678.
107. Amores, J. Multiple instance classification: Review, taxonomy and comparative study / J. Amores // *Artificial Intelligence*. – 2013. – Vol. 201. – P. 81–105.
108. Kotsiantis, S. B. Supervised machine learning: A review of classification techniques / S. B. Kotsiantis, I. Zaharakis, P. Pintelas // *Emerging artificial intelligence applications in computer engineering*. – 2007. – Vol. 160. – P. 3–24.
109. Do we need hundreds of classifiers to solve real world classification problems? / M. Fernández-Delgado, et al. // *The Journal of Machine Learning Research*. – 2014. – Vol. 15.1. – P. 3133–3181.
110. The great time series classification bake off: a review and experimental evaluation of recent algorithmic advances / A. Bagnall, et al. // *Data Mining and Knowledge Discovery*. – 2017. – Vol. 31.3. – P. 606–660.
111. Deep learning for time series classification: a review / H. I. Fawaz, et al. // *Data Mining and Knowledge Discovery*. – July 2019. – Vol. 33, Iss. 4. P. 917–963.
112. The UEA multivariate time series classification archive / A. Bagnall, H. A. Dau, J. Lines, M. Flynn, J. Large, A. Bostrom, E. Keogh, et al. – 2018 [Электронный ресурс]. – Режим доступа: <https://arxiv.org/pdf/1811.00075>
113. The UCR time series archive / H. A. Dau, A. Bagnall, K. Kamgar, C. C. M. Yeh, Y. Zhu, S. Gharghabi, E. Keogh, et al. – 2018 [Электронный ресурс]. – Режим доступа: <https://arxiv.org/pdf/1810.07758>
114. Sirisambhand, K. A Dimensionality Reduction Technique for Time Series Classification Using Additive Representation. / K. Sirisambhand, C. A. Ratanamahatana // *Third*

International Congress on Information and Communication Technology. – Singapore: Springer, 2019. – P. 717–724.

115. A comparison of time series similarity measures for classification and change detection of ecosystem dynamics / S. Lhermitte, et al. // *Remote Sensing of Environment*. – 2011. – Vol. 115.12. – P. 3129–3152.

116. Chapman, A. D. Principles of data quality / A. D. Chapman. – GBIF, 2005.

117. Gabidulin, E. M. Lectures on information theory / E. M. Gabidulin, N. I. Pilipchuk. – MIPT, 2007.

118. Dash, M. Feature selection for classification / M. Dash, H. Liu // *Intelligent data analysis*. – 1997. – Vol. 1(3). – P. 131–156.

119. Automatic classifier selection for non-experts / M. Reif, F. Shafait, M. Goldstein, T. Breuel, A. Dengel // *Pattern Analysis and Applications*. – 2014. – Vol. 17(1). – P. 83–96.

120. Zinnikus, I. An Ontology-Based Recommendation System / I. Zinnikus, C. Fraser, K. Fischer. – *Proceedings of Artificial Intelligence and Applications*, 2004.

121. Bernstein, A. Toward intelligent assistance for a data mining process: An ontology-based approach for cost-sensitive classification / A. Bernstein, F. Provost, S. Hill // *IEEE Transactions on knowledge and data engineering*. – 2005. – Vol. 17 (4). – P. 503–518.

122. Anastácio, I. Supervised Learning for Linking Named Entities to Knowledge Base Entries / I. Anastácio, B. Martins, P. Calado. – TAC, 2011.

123. Ontology-based meta-mining of knowledge discovery workflows / M. Hilario, P. Nguyen, H. Do, A. Woznica, A. Kalousis // *Meta-learning in computational intelligence*. – Springer-Verlag, 2011. – P. 273–315.

124. Panov, P. Ontology of core data mining entities / P. Panov, L. Soldatova, S. Džeroski // *Data Mining and Knowledge Discovery*. – 2014. – Vol. 28(5–6). – P. 1222–1265.

125. Exploring Reasoning with the DMOP Ontology / C. M. Keet, C. d'Amato, Z. C. Khan, A. Lawrynowicz // *Proceedings of ORE*. – 2014. – Vol. 1207. – P. 64–70.

126. Ontology-Based Platform for Conceptual Guided Dataset Analysis / M. Á. Rodríguez-García, J. Medina-Moreira, K. Lagos-Ortiz, H. Luna-Aveiga, F. García-Sánchez, R. Valencia-García // *Distributed Computing and Artificial Intelligence*, 13th International Conference. – Springer, Cham., 2016. – P. 155–163.

127. Automatic classifier selection for non-experts / M. Reif, F. Shafait, M. Goldstein, T. Breuel, A. Dengel // *Pattern Analysis and Applications*. – 2014. – Vol. 17(1). – P. 83–96.

128. Paulheim, H. Machine Learning with and for Semantic Web Knowledge Graphs / H. Paulheim // *Reasoning Web. Learning, Uncertainty, Streaming, and Scalability*. – Springer, Cham., 2018. – P. 110–141.

129. Ristoski, P. Semantic Web in data mining and knowledge discovery: A comprehensive survey. / P. Ristoski, H. Paulheim // *Web Semantics: Science, Services and Agents on the World Wide Web*. – 2016. – Vol. 36. – P. 1–22.
130. Fayyad, U. From data mining to knowledge discovery in databases / U. Fayyad, G. Piatetsky-Shapiro, P. Smyth // *AI Magazine*. – 1996. – Vol. 17 (3). – P. 37.
131. A survey of intelligent assistants for data analysis / F. Serban, J. Vanschoren, J.-U. Kietz, A. Bernstein // *ACM Computing Surveys (CSUR)*. – 2013. – Vol. 45 (3). – P. 31.
132. Abu-Salih, B. Twitter mining for ontology-based domain discovery incorporating machine learning / B. Abu-Salih, P. Wongthongtham, K. Y. Chan // *Journal of Knowledge Management*. – 2018. – Vol. 22 (5). – P. 949–981.
133. CredSaT: Credibility ranking of users in big social data incorporating semantic analysis and temporal factor / B. Abu-Salih, P. Wongthongtham, K. Y. Chan, D. Zhu // *Journal of Information Science*. – 2019. – Vol. 45 (2). – P. 259–280.
134. State-of-the-art ontology annotation for personalised teaching and learning and prospects for smart learning recommender based on multiple intelligence and fuzzy ontology / P. Wongthongtham, K. Y. Chan, V. Potdar, B. Abu-Salih, S. Gaikwad, P. Jain // *International Journal of Fuzzy Systems*. – 2018. – Vol. 20 (4). – P. 1357–1372.
135. Wongthongtham, P. Ontology-based approach for identifying the credibility domain in social Big Data / P. Wongthongtham, B. Abu-Salih // *Journal of Organizational Computing and Electronic Commerce*. – 2018. – Vol. 28 (4). – P. 354–377.
136. Social Credibility Incorporating Semantic Analysis and Machine Learning: A Survey of the State-of-the-Art and Future Research Directions / B. Abu-Salih, B. Bremie, P. Wongthongtham, K. Duan, T. Issa, K. Y. Chan, M. Alahmari, et al. // *Workshops of the International Conference on Advanced Information Networking and Applications*. – Springer, Cham., 2019. – P. 887–896.
137. Calero, C. *Ontologies for Software Engineering and Software Technology* / C. Calero, F. Ruiz, M. Piattini. – Berlin: Springer-Verlag, 2006. – P. 339.
138. Жукова, Н. А. О возможностях синтеза многоуровневых моделей объектов / Н. А. Жукова // *Известия СПбГЭТУ «ЛЭТИ»*. – 2019. – № 10. – С. 16–21.
139. Gasevic, D. *Model Driven Architecture and Ontology Development* / D. Gasevic, D. Djuric, V. Devedzi. – Berlin; Heidelberg: Springer-Verlag, 2006.
140. Robinson, J. A. A machine – oriented logic based on resolution principle / J. A. Robinson // *Journal of the ACM*. – 1965. – Vol. 12. – P. 23–41.
141. Chang, C. *Symbolic Logic and Mechanical Theorem Proving* / C. Chang, R. Lee. – New York: Academic, 1973.

142. Маслов, С. Ю. Теория дедуктивных систем и ее применения [Текст] = Theory of Deductive Systems and Its Applications / С. Ю. Маслов. – М.: Радио и связь, 1986.
143. Тыгу, Э. Х. Алгоритмы структурного синтеза программ = Algorithms for structural synthesis of programs / Э. Х. Тыгу, М. Я. Харф // Программирование. – 1980. – № 4. – С. 3–13.
144. Korukhova, Yu. An approach to automatic deductive synthesis of functional programs / Yu. Korukhova // Annals of Mathematics and Artificial Intelligence. – 2007. – Vol. 50 (3–4). – P. 255–271.
145. Novoseltsev, V. B. Synthesis of parallel recursive programs in structural functional models / V. B. Novoseltsev // Programming and Computer Software. – 2007. – Vol. 33 (5). – P. 293–299.
146. Giacomo, G. Automatic behavior composition synthesis / G. Giacomo, F. Patrizi, S. Sardina // Artificial Intelligence. – March 2013. – Vol. 196. – P. 106–142.
147. Kreitz, C. Program Synthesis. Ch. III.2.5 of Automated Deduction – A Basis for Application / C. Kreitz. – Kluwer Publ., 1998. – P. 105–134.
148. Avellone, A. Synthesis of Programs in Abstract Data Types / A. Avellone, M. Ferrari, P. Miglioli // Logic-Based Program Synthesis and Transformation. LOPSTR 1998. – Berlin: Springer, 1998. – P. 81–100.
149. Srivastava, S. Template-based program verification and program synthesis / S. Srivastava, S. Gulwani, J. S. Foster // International Journal of Software Tools for Technology Transfer. – 2013. – Vol. 15 (5). – P. 497–518.
150. Tahat, A. A Hybrid Method for the Verification and Synthesis of Parameterized Self-Stabilizing Protocols / A. Tahat, A. Ebneenasir // Logic-Based Program Synthesis and Transformation. LOPSTR 2014 (Canterbury, UK, Sept. 9–11, 2014). – Springer, Cham., 2014. – P. 201–218.
151. Kant, E. On the efficient synthesis of efficient programs / E. Kant // Artificial Intelligence. – May 1983. – Vol. 20 (3). – P. 253–305.
152. A Multi-level Approach to Program Synthesis / W. Bibel, D. Korn, C. Kreitz, F. Kurucz, J. Otten, S. Schmitt, G. Stolpmann // Logic Program Synthesis and Transformation. LOPSTR 1997. – Berlin: Springer, 1997. – P. 1–27.
153. Fu, P. A Type-Theoretic Approach to Resolution / P. Fu, E. Komendantskaya // Logic-Based Program Synthesis and Transformation. LOPSTR 2015. – Berlin: Springer, 2015. – P. 91–106.
154. Gulwani, S. Program Synthesis. Foundations and Trends / S. Gulwani, O. Polozov, R. Singh // Programming Languages. – 2017. – Vol. 4, № 1–2. – P. 1–119.

155. Program Synthesis in 2017–18, 2018 [Электронный ресурс]. – Режим доступа: <https://alexpolozov.com/blog/program-synthesis-2018>
156. Компиляторы: принципы, технологии и инструментарий / А. В. Ахо, М. С. Лам, Р. Сети, Д. Д. Ульман. – 2-е изд. – М.: Вильямс, 2010.
157. Kelly, S. Domain-Specific Modeling / S. Kelly, J. Tolvanen. – Hoboken, New Jersey: John Wiley & Sons, Inc., 2008. – 427 p.
158. Studer, R. Semantic Web Services. Concepts, Technologies, and Applications / R. Studer, S. Grimm, A. Abecker – Berlin: Springer, 2007.
159. Boyer, J. Agile Business Rule Development / J. Boyer, H. Mili. – Berlin; Heidelberg: Springer-Verlag, 2011. – 567 p.
160. Friedman-Hill, E. Jess in Action / E. Friedman-Hill. – Greenwich: Maning Publication Co., 2003. – P. 480.
161. Distributed Technical Object Model Synthesis Based on Monitoring Data / V. Yu. Osipov, A. I. Vodyaho, N. A. Zhukova, M. Tianxing, S. Lebedev // International Journal of Knowledge and Systems Science (IJKSS). – 2019. – Vol. 10, Iss. 3. – Art. 3. – P. 27–43.
162. Automatic Synthesis of Multilevel Automata Models of Biological Objects / V. Osipov, M. Lushnov, E. Stankova, A. Vodyaho, Yu. Shichkina, N. Zhukova // Proceedings of International Conference on Computational Science and Its Applications (ICCSA 2019). – Springer, Cham., 2019. – P. 441–456.
163. Tianxing, M. A Knowledge-Oriented Recommendation System for Machine Learning Algorithm Finding and Data Processing / M. Tianxing, N. Zhukova etc. // International Journal of Embedded and Real-Time Communication Systems (IJERTCS). – 2019. – Vol. 10, Iss. 4. – Art. 2. – P. 20–38.
164. About One Approach to Multilevel Behavioral Program Synthesis for Television Devices / V. Yu. Osipov, N. A. Zhukova, A. I. Vodyaho // International Journal of Computers and Communications. – 2017. – Vol. 11. – P. 17–25.
165. Multilevel Automatic Synthesis of Behavioral Programs for Smart Devices / V. Osipov, A. Vodyaho, N. Zhukova, P. Glebovsky // Control, Artificial Intelligence, Robotics & Optimization (ICCAIRO 2017). – IEEE, 2017. – P. 335–340.
166. Bhatti, H. R. Automatic Measurement of Source Code Complexity: Master's Thesis / H. R. Bhatti. – Lulea University of Technology, Lulea, Sweden, 2011.
167. Харари, Ф. Теория графов / Ф. Харари. – М.: Editorial URSS, 2003. – С. 296.
168. Оре, О. Теория графов / О. Оре. – 2-е изд. – М.: Наука, Глав. ред. физ.-мат. лит-ры, 1980. – С. 336.

169. Осипов, В. Ю. Автоматический синтез программ действий интеллектуальных роботов / В. Ю. Осипов // Программирование. – 2016. – № 3. – С. 47–54.
170. Осипов, В. Ю. Синтез результирующих программ управления информационно-вычислительными ресурсами / В. Ю. Осипов // Приборы и системы управления. – 1998. – № 12. – С. 24–27.
171. Радиоэлектронная борьба: теоретические основы / В. Ю. Осипов, А. П. Ильин, В. П. Фролов, А. П. Кондратюк. – Петродворец: ВМИРЭ, 2006.
172. Osipov, V. Yu. Automatic synthesis of action programs for intelligent robots / V. Yu. Osipov // Programming and Computer Software. – 2016. – Vol. 42, Iss. 3. – P. 155–160.
173. Mathematical Methods of Biological System Models Synthesis from the Medical Data / M. S. Kupriyanov, A. I. Vodyaho, N. A. Zhukova, D. I. Kurapeev, M. S. Lushnov, V. Yu. Osipov // Northwest Russia conference on mathematical methods in engineering and technology (MMET NW) (Saint-Petersburg, Sept. 10-14, 2018). – IEEE, 2018. – P. 512–515.
174. Inductive Synthesis of the Models of Biological Systems According to Clinical Trials / V. Osipov, M. Lushnov, E. Stankova, A Vodyaho // International Conference on Computational Science and Its Applications (ICCSA 2017). Lecture Notes in Computer Science. – Springer, Cham., 2017. – Vol. 10404. – P. 103–115.
175. Синтез моделей природных объектов по данным наблюдений / В. Ю. Осипов, А. И. Водяхо, А. Б. Тристанов, Н. А. Жукова, О. О. Луковенкова // Вестник КРАУНЦ. Физ.-мат. науки. – 2018. – № 4 (24). – С. 1–12.
176. Метод трансформации данных для моделирования технических объектов / Н. А. Жукова, А. И. Водяхо, Д. К. Левоневский, А. С. Симоненко // XXI Междунар. конф. по мягким вычислениям и измерениям (SCM-2018) (СПб., 23–25 мая 2018 г.): Сб. докл. в 2 т. – СПб.: СПбГЭТУ «ЛЭТИ», 2018. – С. 450–454.
177. Жукова, Н. А. Онтологические модели трансформации данных о состоянии технических объектов / Н. А. Жукова // Онтология проектирования. – 2019. – Т. 9. – № 3 (33). – С. 345–360.
178. Срагович, В. Г. Адаптивное управление / В. Г. Срагович. – М.: Наука, 1981. – 384 с.
179. Юдин, Д. Б. Математические методы управления в условиях неполной информации / Д. Б. Юдин. – М.: Сов. радио, 1974.
180. Zhukova, N. A. A Formal Framework for Data Fusion / N. A. Zhukova, I. R. Baymuratov // International Journal of Applied Mathematics and Informatics. – 2017. – Vol. 11. – P. 56–64.

181. Baymuratov, I. Logical and Mathematical Models of Data Fusion / I. Baymuratov, N. Zhukova // Control, Artificial Intelligence, Robotics & Optimization (ICCAIRO 2017). –IEEE, 2017. – P. 121–126.

182. Баймуратов, И. Р. О логическом подходе в теории информации / И. Р. Баймуратов, Н. А. Жукова // Двадцать пятая международная конференция «Математика. Компьютер. Образование» (г. Дубна, 29 янв. – 3 февр. 2018 г.). [Электронный ресурс]. – Режим доступа: <http://www.mce.su/rus/archive/abstracts/mce25/authors/person283231/doc311959/>

183. Baimuratov, I. R. Verification of the Formal Approach to Data Fusion / I. R. Baimuratov, N. A. Zhukova, S. D. Morozov // Supplementary Proceedings of the 7th International Conference on Analysis of Images, Social Networks and Texts (AIST 2018) // CEUR Workshop Proceedings. – 2018. – Vol. 2268. – P. 179–189.

184. Baymuratov, I. R. An Approach to Clustering Models Estimation / I. R. Baimuratov, N. A. Zhukova // Proceedings of the 22nd Conference of Open Innovations Association FRUCT (Jyvaskyla, Finland, May 15–18, 2018) / Ed. by S. Balandin, T. Hämmäläinen, T. Tyutina. – Helsinki: FRUCT Oy, 2018. – P. 19–24. – ISSN 2305-7254. – ISBN 978-952-68653-4-8.

185. The Information Evaluation System of Data Processing Results / N. Zhukova, I. Baymuratov, Nguyen Than, N. Mustafin // Proceedings of the 24th Conference of Open Innovations Association FRUCT (Moscow, Russia, April 08–12, 2019). – 2019. – Art. 117. – С. 810–814.

186. A Bayesian Information Criterion for Unsupervised Learning Based on an Objective Prior / I. Baimuratov, Yu. Shichkina, E. Stankova, N. Zhukova, Nguyen Than // Proceedings of International Conference on Computational Science and Its Applications (ICCSA 2019). – Springer, Cham., 2019. – P. 707–716.

187. Tianxing, M. An Ontology of Machine Learning Algorithms for Human Activity Data Processing / M. Tianxing, N. Zhukova // Proceedings of the 22nd Conference of Open Innovations Association FRUCT (Jyvaskyla, Finland, May 15–18, 2018) / Ed. by S. Balandin, T. Hämmäläinen, T. Tyutina. – Helsinki: FRUCT Oy, 2018. – P. 392–396. – ISSN 2305-7254. – ISBN 978-952-68653-4-8.

188. Tianxing, M. A Knowledge-based Recommendation System for Time Series Classification / M. Tianxing, N. Zhukova, N. Mustafin // Proceedings of the 24th Conference of Open Innovations Association FRUCT (Moscow, Russia, April 08–12, 2019). – 2019. – Art. 24. – P. 751–759.

189. Knowledge-Based Computational Environment for Real-World Data Processing / M. Tianxing, N. Zhukova, V. Meltsov, Y. A. Shichkina // International Conference on Computational



Science and Its Applications. Lecture Notes in Computer Science. – Springer, Cham., 2017. – Vol. 10404. – P. 257–269.

190. Tianxing, M. A Multi-layer Ontology for Data Processing Techniques / M. Tianxing, N. Zhukova, etc. // The proceeding of the 16th International Conference on Informatics in Control, Automation and Robotics (ICINCO 2019) (Prague, Czech Republic, July 29–31, 2019). – SciTePress, 2019. – P. 648–655.

191. Витол, А. Д. Технология адаптивной обработки измерительных данных [Текст]: Монография / А. Д. Витол, Н. А. Жукова. – СПб.: Изд-во СПбГЭТУ «ЛЭТИ», 2012.

192. Когнитивный мониторинг телекоммуникационных сетей [Текст]: Монография / В. Ю. Осипов, А. Калмацкий, А. И. Водяхо, Н. А. Жукова, П. А. Глебовский. – СПб.: Изд-во СПбГЭТУ «ЛЭТИ», 2018. – 202 с.

193. Modeling Framework for Medical Data Semantic Transformations / A. Vodyaho, N. Zhukova, D. Kurapeev, M. Lushnov // Workshop on Data Analysis in Medicine (WDAM-2017). – 2017. – Vol. 6. – P. 67–82.

194. Vitol, A. D. Model for knowledge representation of multidimensional measurements processing results in the environment of intelligent GIS / A. D. Vitol, A.V. Pankin, N. A. Zhukova // Proceedings of the 20th International Conference on Conceptual Structures (India, Mumbai, Jan. 10–12, 2013). – Berlin; Heidelberg: Springer, 2013. – P. 266–276.

195. Метод идентификации групповых телеметрических сигналов на основе частотно-рангового распределения / В. Э. Балтрашевич, А. В. Васильев, Н. А. Жукова, И. С. Соколов // 14-ая Всерос. конф. «Математические методы распознавания образов» (Суздаль, 21–26 сент. 2009 г.): Сб. докл. – М.: МАКС Пресс, 2009. – С. 305–308.

196. Балтрашевич, В. Э. Графовая модель группового телеметрического сигнала со сменой кадра / В. Э. Балтрашевич, Н. А. Жукова, И. С. Соколов // Известия СПбГЭТУ «ЛЭТИ». Сер. Информатика, управление и компьютерные технологии. – 2010. – Вып. 10. – С. 12–17.

197. Жукова, Н. А. Метод восстановления структуры группового телеметрического сигнала на основе графовой модели / Н. А. Жукова, И. С. Соколов // Труды СПИИРАН. – 2010. – № 2. – С. 45–66.

198. Oreshko, N. I. Segmentation and cluster analysis of piecewise constant random processes on the base of spectral characteristics by using wavelet technology / N. I. Oreshko, A. V. Ecalo, N. A. Zhukova // Proceedings of 8th International Conference on Pattern Recognition and Image Analysis: New Information Technologies (Yoshkar-Ola, Mari State Technical University, Oct 8–12, 2007). – 2007. – Vol. 2. – P. 388–391.

199. Жукова, Н. А. Сегментация кусочно-постоянных случайных процессов по спектральным характеристикам на основе нечеткой кластеризации и вейвлет-технологии / Н. А. Жукова, Н. И. Орешко // Десятая междунар. конф. и выставка «Цифровая обработка сигналов и ее применение» (М., 26–28 марта 2008 г.): Сб. докл. – М.: Изд-во ИПУ, 2008. – Вып. X-1. – С. 408–412.
200. Витол, А. Д. Алгоритм сегментации кусочно-постоянных телеметрических параметров / А. Д. Витол, А. Б. Тристанов, Н. А. Жукова // Вестник КРАУНЦ. Физ.-мат. науки. – 2010. – № 1 (1). – С. 46–53.
201. Жукова, Н. А. Метод определения типов медленно меняющихся телеметрических параметров с использованием редакционного расстояния и символьного представления / Н. А. Жукова, И. С. Соколов // Сборник научных трудов НГТУ. – 2012. – № 1 (67). – С. 87–94.
202. Васильев, А. В. Применение методов распознавания образов в задаче контроля состояния сложных технических объектов / А. В. Васильев, А. О. Дерипаска, Н. А. Жукова // Известия СПбГЭТУ «ЛЭТИ». Сер. Информатика, управление и компьютерные технологии. – 2010. – Вып. 10. – С. 88–93.
203. Жукова, Н. А. Гармонизация, интеграция и слияние многомерных измерений параметров природных и технических объектов в системах мониторинга / Н. А. Жукова // Известия СПбГЭТУ «ЛЭТИ». Сер. Информатика, управление и компьютерные технологии. – 2014. – Вып. 2. – С. 27–33.
204. Vodyaho, A. Implementation of JDL Model for multidimensional measurements processing in the environment of intelligent GIS / A. Vodyaho, N. Zhukova // International Journal of Conceptual Structures and Smart Applications. – 2014. – Vol. 1, № 3. – P. 36–56.
205. Дерипаска, А. О. Адаптивный выбор процессов обработки и анализа многомерных измерений в интеллектуальных информационных системах / А. О. Дерипаска, Н. А. Жукова, А. В. Панькин // Тринадцатая нац. конф. по искусственному интеллекту с междунар. участием (Белгород, 16–20 окт. 2012 г.): Сб. тр. – Белгород: Изд-во БГТУ, 2012. – Т. 1. – С. 181–189.
206. Vitol, A. D. Adaptive multidimensional measurements processing using IGIS technologies / A. D. Vitol, A.V. Pankin, N. A. Zhukova // Proceedings of the 6th International Workshop on Information Fusion and Geographic Information Systems: Environmental and Urban Challenges (Saint-Petersburg, May 12–15, 2013). – Berlin; Heidelberg: Springer, 2014. – P. 179–200.
207. Геппенер, В. В. Принципы организации поддержки принятия решений в адаптивных интеллектуальных геоинформационных системах / В. В. Геппенер, Н. А. Жукова, А. В. Панькин // Междунар. конф. по мягким вычислениям и измерениям

(СПб., 23–25 мая 2013 г.): Сб. докл. – СПб.: Изд-во СПбГЭТУ «ЛЭТИ», 2013. – Т. 1. – С. 183–188.

208. Vodyaho, A. I. Simulation of Processing and Analyses of Telemetry Information about the State of Complex Dynamic Objects / A. I. Vodyaho, N. A. Zhukova // Proceedings of International Conference on Computer Technologies in Physical and Engineering Applications (Saint-Petersburg, 30 June – 4 July, 2014). – IEEE, 2014. – P. 206–207.

209. Водяхо, А. И. Модель адаптивной обработки и анализа многомерных измерений / А. И. Водяхо, Н. А. Жукова // Двадцать вторая международная конференция «Математика. Компьютер. Образование» (г. Пущино, 26–31 янв. 2015 г.). [Электронный ресурс]. – Режим доступа: <http://www.mce.su/archive/doc216000/rus.pdf>

210. Service-Oriented Architecture of Intelligent GIS / A. V. Pankin, V. V. Popovich, S. N. Potapichev, F. R. Galiano, N. A. Zhukova // Proceedings of International Symposium on Service-Oriented Mapping (Austria, Vienna, Nov. 22–23, 2012). – JOBSTMedia, GmbH, 2012. – P. 297–311.

211. Semantic Platform for building coherent net of smart services / A. Vodyaho, N. Zhukova, M. Kolchin, M. Lapaev // REAL CORP 2016 – SMART ME UP! How to become and how to stay a Smart City, and does this improve quality of life? Proceedings of 21st International Conference on Urban Planning, Regional Development and Information Society. – 2016. – P. 561–568 [Электронный ресурс]. – Режим доступа: <http://repository.corp.at/133/>

212. Жукова, Н. А. Принципы организации управления процессами обработки и анализа многомерных измерений в ИГИС / Н. А. Жукова, А. В. Панькин // Материалы 5-й Рос. мультikonф. по проблемам управления «Информационные технологии в управлении» (ИТУ-2012) (СПб., 9–11 окт. 2012 г.). – СПб.: АО Концерн «ЦНИИ „Электроприбор”», 2012. – С. 403–414.

213. Zhukova N. Dynamic Resources Management in Agile IGIS / N. Zhukova // Information Fusion and Geographic Information Systems (IF&GIS' 2015): 7th International Workshop on Information Fusion and Geographic Information Systems: Deep Virtualization for Mobile (Grenoble, France, May 18–20, 2015) / Ed. by V. Popovich, C. Claramunt, M. Schrenk, K. Korolenko, J. Gensel. – Springer International Publishing, 2015. – P. 125–145. (Lecture Notes in Geoinformation and Cartography).

214. Вычислительные модели когнитивных систем мониторинга / А. И. Водяхо, Н. А. Жукова, Н. В. Климов, О. О. Луковенкова, В. Ю. Осипов, А. Б. Тристанов // Морские интеллектуальные технологии. – 2018. – Т. 3. – № 4 (42). – С. 147–153.

215. Computational and technological models of cognitive monitoring systems / V. U. Osipov, A. I. Vodyaho, N. V. Klimov, N. A. Zhukova, M. A. Chervontsev // *Advances in Science, Technology and Engineering Systems Journal*. – 2019. – Vol. 2, Iss. 1. – P. 197–202.
216. Vodyaho, A. System of Ontologies for Data Processing Applications Based on Implementation of Data Mining Techniques / A. Vodyaho, N. Zhukova // *Proceedings of the 3rd International Conference on Analysis of Images, Social Networks and Texts, AIST 2014 (Yekaterinburg, Russia, April, 2014)*. – 2014. – Vol. 1197. – P. 102–116.
217. Модели представления данных в области медицины / Д. А. Коробов, М. В. Лапаев, А. И. Водяхо, Н.А. Жукова // *Известия СПбГЭТУ «ЛЭТИ»*. – 2016. – № 7. – С. 7–13.
218. Водяхо, А. И. Онтологический подход к построению систем мониторинга ресурсов в сетях кабельного телевидения / А. И. Водяхо, Н. Г. Мустафин, Н. А. Жукова // *Известия СПбГЭТУ «ЛЭТИ»*. – 2017. – № 2. – С. 29–38.
219. Feng, S. Classification of Curves in 2D and 3D via Affine Integral Signatures / S. Feng, I. A. Kogan, H. Krim // *Acta Applicandae Mathematicae*. – March 2010. – Vol. 109, Iss. 3. – P. 903–937.
220. Chang, K. Principal curves for nonlinear feature extraction and classification / K. Chang, J. Ghosh // *SPIE Applications of Artificial Neural Networks in Image Processing III*. – 1998. – Vol. 3307. – P. 120–129.
221. Hallin, M. Rank tests for time series analysis: a survey / M. Hallin, M. Puri // *New Directions in Time Series Analysis*. – New York: Springer-Verlag, 1992. – P. 111–154.
222. Gevins, A. Statistical pattern recognition. Handbook of electroencephalography and clinical neurophysiology: methods of brain and magnetic signals / A. Gevins, A. Remond. – Vol. 1. – Amsterdam: Elsevier, 1987.
223. Gevins, A. Correlation analysis. Handbook of electroencephalography and clinical neurophysiology: methods of brain and magnetic signals / A. Gevins, A. Remond. – Vol. 1. – Amsterdam: Elsevier, 1987.
224. Бокс, Дж. Анализ временных рядов прогноз и управление / Дж. Бокс, Г. Дженкинс. – Т. 1–2. – М.: Мир, 1974.
225. Cellucci, C. Statistical validation of mutual information calculations: Comparison of alternative numerical algorithms / C. Cellucci, A. Albano, P. Rapp // *Physical Review*. – 2005. – E 71: 066208.
226. Assessment of Measures of Scalar Time Series Analysis in Discriminating Preictal States. / D. Kugiumtzis, I. Vlachos, A. Papan, P.G. Larsson // *International Journal of Bioelectromagnetism*. – 2007. – Vol. 9, № 3. – P. 134–145.

227. Hinich, M. Testing for Dependence in the Input to a Linear Time Series Model / M. Hinich // *Journal of Nonparametric Statistics*. – 1996. – Vol. 6, Iss. 2–3. – P. 205–221.
228. Berkeley Detecting Nonlinearity in Time Series: Surrogate and Bootstrap Approaches / M. Hinich, M. Eduardo, E. Mendes, L. Stone // *Studies in Nonlinear Dynamics & Econometrics*. – 2005. – Vol. 9 (4). – P. 1–15.
229. Grassberger, P. Estimation of the Kolmogorov entropy from a chaotic signal / P. Grassberger, I. Procaccia // *Physical Review*. – 1983. – Vol. A28. – P. 2591.
230. Improved spatial characterization of the epileptic brain by focusing on nonlinearity / R. G. Andrzejak, F. Mormann, G. Widman, T. Kreuz, C. E. Elger, K. Lehnertz // *Epilepsy Research*. – 2006. – Vol. 69. – P. 30–44.
231. McSharry, P. E. Comparison of predictability of epileptic seizures by a linear and a nonlinear method / P. E. McSharry, L. A. Smith, L. Tarassenko // *IEEE Transactions on Biomedical Engineering*. – 2003. – Vol. 50 (5). – P. 628–633.
232. Kennel, M. Determining embedding dimension for phase-space reconstruction using a geometrical construction / M. Kennel, R. Brown, H. D. I. Abarbanel // *Physical Review*. – 1992. – Vol. A45. – P. 3403–3411.
233. Pincus, S. M. Approximate entropy as a measure of system complexity / S. M. Pincus // *Proceedings of the National Academy of Sciences of USA*. – 1991. – Vol. 88. – P. 2297–2301.
234. Radhakrishnan, N. An alternate partitioning technique to quantify the regularity of complex time series / N. Radhakrishnan, W. D. James, P. C. Loizou // *International Journal of Bifurcation and Chaos*. – 2000. – Vol. 10 (7). – P. 1773–1779.
235. Identifying deterministic signals in simulated gravitational wave data: Algorithmic complexity and surrogate data method / Y. Zhao, M. Small, D. Coward, E. Howell, C. N. Zhao, L. Ju, D. G. Blair // *Classical and Quantum Gravity*. – 2006. – Vol. 23. – P. 1801–1814.
236. Farmer, J. D. Predicting chaotic time series / J. D. Farmer, J. J. Sidorowich // *Physical Review Letters*. – 1987. – Vol. 59. – P. 845–848.
237. Kugiumtzis, D. State space local linear prediction / D. Kugiumtzis // *Modelling and Forecasting Financial Data, Techniques of Nonlinear Dynamics* / Ed. by A. Soofi, L. Cao. – Kluwer Academic Publishers, 2002. – Ch. 4. – P. 95–113.
238. Hjorth, B. EEG analysis based on time domain properties / B. Hjorth. // *Electroencephalography and Clinical Neurophysiology*. – 1970. – Vol. 29. – P. 306–310.
239. Edgar, E. P. *Chaos and Order in the Capital Markets: A New View of Cycles, Prices, and Market Volatility* / E. P. Edgar. – 2nd ed. – New York: John Wiley & Sons, 1996.

240. Mosaic organization of DNA nucleotides / C.-K. Peng, S. V. Buldyrev, S. Havlin, M. Simons, H. E. Stanley, A. L. Goldberger // *Physical Review E*. – 1994. – Vol. 49 (2). – P. 1685–1689.
241. Kugiumtzis, D. Measures of Analysis of Time Series (MATS): A MATLAB Toolkit for Computation of Multiple Measures on Time Series Data Bases / D. Kugiumtzis, A. Tsimpiris // *Journal of Statistical Software*. – 2010. – Vol. 33 (5). – P. 1–30.
242. Dunn, J. Well separated clusters and optimal fuzzy partitions / J. Dunn // *Journal of Cybernetics*. – 1974. – Vol. 4, № 1. – P. 95–104.
243. Davies, D. A cluster separation measure / D. Davies, D. Bouldin // *IEEE PAMI*. – 1979. – Vol. 1, № 2. – P. 224–227.
244. Sharma, S. Applied multivariate techniques / S. Sharma. – New York: John Wiley & Sons, Inc., 1996.
245. Rousseeuw, P. Silhouettes: a graphical aid to the interpretation and validation of cluster analysis / P. Rousseeuw // *Journal of Computational and Applied Mathematics*. 1987. – Vol. 20, № 1. – P. 53–65.
246. Liu, Y. Understanding of Internal Clustering Validation Measures / Y. Liu, Z. Li, et al. // *Proceeding ICDM'10 Proceedings of the 2010 IEEE International Conference on Data Mining (Sydney, Australia, Dec. 13–17, 2010)*. – 2010. – P. 911–916.
247. Vodyaho, A. I. Implementation of Agile Concepts in Recommender Systems for Data Processing and Analyses / A. I. Vodyaho, N. A. Zhukova // *Communications in Computer and Information Science*. Vol. 542: Analysis of Images, Social Networks and Texts. 4th International Conference, AIST 2015 (Yekaterinburg, Russia, April 9–11, 2015). – Springer, 2015. – P. 443–457.
248. Vodyaho, A. Conceptual Model for Routine Measurements Processing and Analyses in Adaptive Intelligent Information Systems / A. Vodyaho, N. Zhukova, M. Lapaev // *Supplementary Proceedings of the Fifth International Conference on Analysis of Images, Social Networks and Texts (AIST'2016) (Yekaterinburg, Russia, April, 2016) // CEUR Workshop Proceeding*. – 2016. – Vol. 1710. – P. 357–369.
249. Лебедев, С. В. Слияние медицинских данных на основе онтологий / С. В. Лебедев, Н. А. Жукова // *Онтологии проектирования*. – 2017. – Т. 7, № 2 (24). – С. 145–159.
250. An Ontology-driven Toolset for Fast Prototyping of Medical Data Processing Systems / S. Lebedev, A. Vodyaho, N. Zhukova, D. Kurapeev, M. Lushnov // *International Journal of Biology and Biomedical Engineering*. – 2017. – Vol. 11. – P. 135–142.
251. Agile Software Architecture / M. A. Babar, A. W. Brown, I. Mistrik. – Waltham, MA: Elsevier Inc., 2014. – P. 392.

252. Bloomberg, J. The agile architecture revolution / J. Bloomberg. – Hoboken, New Jersey: John Wiley & Sons, Inc., 2013. – 280 p.
253. Beck, F. A. Dynamic Reconfigurable Architectures and Transparent Optimization Techniques / F. A. Beck, L. Carro. – Springer, 2010. – 176 p.
254. Tarvainen, P. Adaptability Evaluation at Software Architecture Level / P. Tarvainen // The Open Software Engineering Journal. – 2008. – Vol. 2. – P. 1–30.
255. ISO 9126 (ГОСТРИСО/МЭК 9126–93). Информационная технология. Оценка программного продукта. Характеристики качества и руководство по их применению [Текст]. – Введ. 1994-06-30. – М.: Стандартинформ, 1994. – 12 с.
256. An Architecture-based Approach to Self-adaptive Software / P. Oreizy, M. M. Gorlick, R. N. Taylor, D. Heimbigner, G. Johnson, N. Medvidovc, A. Quilici, D. S. Rosenblum, A. L. Wolf // IEEE Intelligent Systems and their Applications. – 1999. – Vol. 14, № 3. – P. 54–62.
257. Matinlassi, M. The impact of maintainability on component-based software systems / M. Matinlassi, E. Niemelä // Proceedings of the 29th EUROMICRO Conference (EUROMICRO'03) "New Waves in System Architecture". – 2003. – P. 25–32.
258. Aksit, M. Dynamic, adaptive and reconfigurable systems overview and prospective vision / M. Aksit, Z. Choukair // Proceedings of the 23rd International Conference on Distributed Computing Systems Workshops. – 2003. – P. 84–89.
259. Lemos, de R. A co-operative object-oriented architecture for adaptive systems / R. de Lemos // Proceedings of the 7th IEEE International Conference and Workshop on Engineering of Computer Based Systems (ECBS 2000). – 2000. – P. 120–128.
260. Stellman, A. Learning Agile. Understanding Scrum, XP, Lean, and Kanban / A. Stellman, J. Greene. – Sebastopol, CA: O'Reilly, 2014. – 390 p.
261. Водяхо, А. И. Архитектурный подход к построению адаптивных интеллектуальных систем анализа многомерных измерений параметров пространственно соотнесенных объектов [Текст]: Монография / А. И. Водяхо, Н. А. Жукова. – СПб.: Изд-во СПбГЭТУ «ЛЭТИ», 2014. – 228 с.
262. Жукова, Н. А. Архитектурное проектирование систем мониторинга состояния сложных технических и природных систем / Н. А. Жукова, А. И. Водяхо, М. А. Червонцев // Известия СПбГЭТУ «ЛЭТИ». – 2018. – № 7. – С. 38–46.
263. Когнитивные технологии в управлении мониторингом / А. И. Водяхо, В. Ю. Осипов, Н. А. Жукова, М. А. Червонцев // Научно-техническая информация. Сер. 2, Информационные процессы и системы. – 2019. – № 4. – С. 1–12.

264. Cognitive Technologies in Monitoring Management / A. I. Vodyaho, V. U. Osipov, N. A. Zhukova, M. A. Chervontsev // *Automatic Documentation and Mathematical Linguistics*. – 2019. – Vol. 53, Iss. 2. – P. 71–80.
265. Cognitive systems for monitoring: Architectural view / M. Chervontsev, A. Vodyaho, N. Zhukova, A. Ekalo, E. Postnikov, V. Osipov // *Advances in Science, Technology and Engineering Systems Journal*. – 2019. – Vol. 4, Iss. 3. – P. 117–125.
266. Мультимодельный подход к построению систем мониторинга / А. И. Водяхо, Н. А. Жукова, С. А. Аббас, М. А. Червонцев // *Известия СПбГЭТУ «ЛЭТИ»*. – 2019. – № 7. – С. 5–13.
267. Software Architecture Knowledge Management / Ed. by M. A. Babar, T. Dingsøyr, P. Lago, H. van Vliet. – Springer, 2009. – P. 279.
268. Водяхо, А. И. Интеграция технологий и наследование знаний как средства повышения эффективности процесса разработки информационных систем / А. И. Водяхо, Н. А. Жукова, М. Г. Пантелеев // *Известия СПбГЭТУ «ЛЭТИ»*. – 2010. – № 5. – С. 61–66.
269. Friedlander, D. S. Semantic information fusion for coordinated signal processing in mobile sensor networks. / D. S. Friedlander, S. Phoha // *International Journal of High Performance Computing Applications*. – 2002. – Vol. 16 (3). – P. 235–241.
270. Blasch, E. High-Level Information Fusion Management and System Design / E. Blasch, E. Bosse, D. Lambert. – Norwood, MA: Artech House Publishers, 2012.
271. Calero, C. Ontologies for Software Engineering and Software Technology / C. Calero, F. Ruiz, M. Piattini. – Springer-Verlag, 2006.
272. Erl, T. SOA: principles of service design / T. Erl. – New York: Prentice Hall, 2008. – 545 p.
273. Frankel, D. S. Model Driven Architecture / D. S. Frankel. – Indianapolis, Indiana: Wiley Publishing, Inc., 2003. – P. 355.
274. Hwang, K. Distributed and Cloud Computing Waltham / K. Hwang, G. C. Fox, J. J. Dongarra. – Elsevier Inc., 2012. – 631 p.
275. International Standard ISO/IEC/IEEE 42010 Systems and software engineering – Architecture description [Электронный ресурс]. – Режим доступа: <http://www.iso.org>
276. Онтологический подход к проектированию проблемно-ориентированных информационных систем / А. И. Водяхо, Н. А. Жукова, М. Г. Пантелеев, Д. В. Пузанков // *Известия СПбГЭТУ «ЛЭТИ»*. – 2010. – № 5. – С. 46–51.
277. Жукова, Н. А. Архитектурный подход к построению систем обработки многомерных измерений параметров пространственно распределенных объектов / Н. А. Жукова, А. И. Водяхо // *Известия СПбГЭТУ «ЛЭТИ»*. – 2013. – № 2. – С. 21–27.



278. Towards Building of Cable TV Content-Sensitive Adaptive Monitoring and Management Systems / V. Yu. Osipov, N. A. Zhukova, A. I. Vodyaho, A. Kalmatsky, N. G. Mustafin // *International Journal of Computers and Communications*. – 2017. – Vol. 11. – P. 75–81. – ISSN 2074-1294.

279. Использование онтологических контекстов в интеллектуальных проблемно-ориентированных информационно-управляющих системах / А. И. Водяхо, Н. А. Жукова, М. Г. Пантелеев, Д. В. Пузанков // *Известия СПбГЭТУ «ЛЭТИ»*. – 2010. – № 9. – С. 46–50.

280. Walton, C. *Agency and the Semantic Web* / C. Walton. – New York: Oxford University Press Inc., 2007.

281. Gasevic, D. *Model Driven Architecture and Ontology Development* / D. Gasevic, D. Djuric, V. Devedzic. – Springer-Verlag, 2006.

282. Архитектурный фреймворк, ориентированный на поддержку процесса разработки многомерных измерений параметров пространственно распределенных объектов. Архитектурный подход к управлению ИТ-сервисами / А. И. Водяхо, С. С. Голяк, С. А. Гордеев, Н. А. Жукова // *Известия СПбГЭТУ «ЛЭТИ»*. – 2013. – № 4. – С. 24–29.

283. Pastor, O. *Model-Driven Architecture in Practice* / O. Pastor, J. C. Molina. – Heidelberg, New York: Springer, 2007. – 302 p.

284. Mellor, S. J. Guest editors' introduction: Model-driven development / S. J. Mellor, A. N. Clark, T. Futagami // *IEEE Software*. – 2002. – Vol. 20, Iss. 5. – P. 14–18.

285. Czarnecki, K. Feature-Based Survey of Model Transformation Approaches / K. Czarnecki, S. Helsen // *IBM Systems Journal*. – 2006. – Vol. 45. – P. 621–645.

286. *Graph and Model Transformation General Framework and Applications* / H. Ehrig, C. Ermel, U. Golas, F. Hermann. – Berlin; Heidelberg: Springer-Verlag, 2015.

287. Bézivin, J. On the Need for Megamodels / J. Bézivin, J. Jouault, P. Valduriez // *Proceedings of the OOPSLA/GPCE: Best Practices for Model-Driven Software Development workshop, 19th Annual ACM Conference on Object-Oriented Programming, Systems, Languages, and Applications*. 2004.

288. Studer, R. *Semantic Web Services. Concepts, Technologies, and Applications* / R. Studer, S. Grimm, A. Abecker – Berlin: Springer, 2007.

289. Taniar, D. *Web semantics and ontology* / D. Taniar, J. W. Rahayu. – London: Idea Group Inc., 2006. – 404 p.

290. *Pattern-Oriented Software Architecture – A System of Patterns* / F. Buschmann, R. Meunier, H. Rohnert, P. Sommerlad, M. Stal. – New York: John Wiley, 1996. – 464 p.

291. Bellifemine, F. *Developing Multi-Agent Systems with JADE* / F. Bellifemine, G. Caire, D. Greenwood. – New Jersey: John Wiley & Sons, 2007.

292. Linden van der, F. *Software Product Lines in Action* / F. van der Linden, K. Schmid, E. Rommes. – Berlin; Heidelberg; New York: Springer, 2007.
293. Эванс, Э. Предметно-ориентированное проектирование. Структуризация сложных программных систем: Пер. с англ. / Э. Эванс. – М.: ИД Вильямс, 2000. – 448 с.
294. Kelly, S. *Domain-Specific Modeling: Enabling Full Code Generation* / S. Kelly, J. Tolvanen. – John Wiley & Sons, 2008. – 340 p.
295. Казнов, Д. В. Методология и инструментарий предметно-ориентированного моделирования / Д. В. Казнов: Автореф. дис. ... д-ра техн. наук. – СПб., 2015. – 32 с.
296. Доменно-онтологический подход к проектированию интеллектуальных систем сбора и обработки телеметрической информации / А. И. Водяхо, Н. А. Жукова, М. Г. Пантелеев, Д. В. Пузанков // Известия высших учебных заведений России. Радиоэлектроника. – 2011. – № 1. – С. 28–35.
297. Duffy, D. *Domain Architectures: Models and Architectures for UML Applications* / D. Duffy. – John Wiley & Sons, 2004.
298. Feature Driven Development (FDD) and Agile Modeling [Электронный ресурс]. – Режим доступа: <http://www.agilemodeling.com/essays/fdd.htm#sthash.laEKmCzj.dpuf>
299. The official website of the United States Department of Defence [Электронный ресурс]. – Режим доступа: <https://www.defense.gov/>
300. The Open Group Architecture Framework [Электронный ресурс]. – Режим доступа: <http://www.opengroup.org/togaf>
301. The Zachman International e-Commerce Site [Электронный ресурс]. – Режим доступа: <https://www.zachman.com/>
302. ISO 15704:2000. Industrial automation systems. Requirements for enterprise-reference architectures and methodologies / ISO. 2000. Appendix: Generalized Enterprise Reference Architecture (GERA).
303. Гринфилд, Д. Фабрики разработки программ: потоковая сборка типовых приложений, моделирование, структуры и инструменты / Д. Гринфилд, К. Шорт. – М.: ИД Вильямс, 2007. – 592 с.
304. Cardoso, J. *Semantic Web Services: Theory, Tools, and Applications* / J. Cardoso. – Hershey, New York: IGI Global, 2007. – 279 p.
305. OWL-S: Semantic Markup for Web Services: Description [Электронный ресурс]. – Режим доступа: <http://www.w3.org/Submission/OWL-S>
306. FIPA [Электронный ресурс]. – Режим доступа: <http://www.fipa.org/>
307. Chappell, D. *Enterprise Service Bus* / D. Chappell. – Sebastopol: O'Reilly Media, 2004. – 328 p.

308. Совместное использование сервисно-ориентированных и многоагентных архитектур для построения интеллектуальных информационно-управляющих систем / А. И. Водяхо, Н. А. Жукова, М. Г. Пантелеев, Д. В. Пузанков // Известия СПбГЭТУ «ЛЭТИ». – 2010. – № 8. – С. 38–43.
309. Kasse, T. Practical Insight into CMMI / T. Kasse. – Boston: Artech House, 2004.
310. Fusion Metrics for Dynamic Situation Analysis / E. Blasch, M. Pribilski, B. Daugherty, B. Roscoe, J. Gunsett // Proceedings of SPIE. – 2004. – Vol. 5429. – P. 428–438.
311. Rozanski, N. Software Systems Architecture: Working With Stakeholders Using Viewpoints and Perspectives / N. Rozanski, E. Woods. – Upper Saddle River, New Jersey: Pearson Education, Inc., 2008. – 546 p.
312. Интеллектуальные географические информационные системы для мониторинга морской обстановки [Текст] / Под общ. ред. чл.-кор. РАН Р. М. Юсупова и д-ра техн. наук В. В. Поповича. – СПб.: Наука, 2013. – 284 с.
313. Когнитивные геоинформационные системы для мониторинга / А. И. Водяхо, В. Ю. Осипов, М. А. Червонцев, Н.А. Жукова // Региональная информатика и информационная безопасность: Сб. тр. Вып. 5 / СПОИСУ. – СПб., 2018. – С. 23–28.
314. Штанько, С. В. Схемы аутентификации данных и пользователей в распределенных информационных системах / С. В. Штанько, Н. А. Жукова // Известия СПбГЭТУ «ЛЭТИ». Сер. Информатика, управление и компьютерные технологии. – 2012. – Вып. 8. – С. 46–51.
315. Штанько, С. В. Схемы аутентификации данных и пользователей в распределенных ГИС / С. В. Штанько, Н. А. Жукова // Известия СПбГЭТУ «ЛЭТИ». Сер. Информатика, управление и компьютерные технологии. – 2012. – Вып. 6. – С. 57–62.
316. Жукова, Н. А. Интеграция программных компонентов корпоративных информационных систем / Н. А. Жукова, А. В. Панькин // Материалы 5-й Рос. мультikonф. по проблемам управления «Информационные технологии в управлении» (СПб., 9–11 окт. 2012 г.). – СПб.: Концерн «ЦНИИ „Электроприбор”», 2012. – С. 414–421.
317. Sparks, G. Enterprise Architect User Guide / G. Sparks; Sparx Systems Pty Ltd. – 2012. – P. 2888 [Электронный ресурс]. – Режим доступа: <https://www.sparxsystems.com.au/bin/EAUUserGuide.pdf>
318. Гамма, Э. Приемы объектно-ориентированного проектирования. Паттерны проектирования / Э. Гамма, Р. Хелм, Джонсон Р. и др. – СПб.: Питер, 2001. – 368 с.
319. Sotomayor, B. Globus Toolkit 4: Programming Java Services / B. Sotomayor, L. Childers. – San Francisco: Elsevier. Inc., 2006. – 506 p.

320. Von Halle, B. *Business Rules Applied: Building Better Systems Using the Business Rules Approach* / B. Von Halle. – New York: Wiley Computer Publishing, 2002. – 592 p.

321. Ньюмен, С. Создание микросервисов / С. Ньюмен. – СПб.: Питер, 2016. – 304 с.

322. Васильев, А. В. Определение семантического состава группового телеметрического сигнала / А. В. Васильев, А. Д. Витол, Н. А. Жукова // Известия СПбГЭТУ «ЛЭТИ». Сер. Информатика, управление и компьютерные технологии. – 2010. – Вып. 10. – С. 18–23.

323. Жукова, Н. А. Метод нечеткого сравнения телеметрических параметров на основе алгоритмов интеллектуального анализа данных / Н. А. Жукова, И. С. Соколов, А. В. Экало // Известия СПбГЭТУ «ЛЭТИ». Сер. Информатика, управление и компьютерные технологии. – 2013. – Вып. 9. – С. 25–29.

324. Pankin, A. V. *Situation Assessment Using Results of Objects Parameters Measurements Analyses in IGIS* / A. V. Pankin, A. I. Vodyaho, N. A. Zhukova // Proceedings of the Workshop “Formal Concept Analysis Meets Information Retrieval” co-located with the 35th European Conference on Information Retrieval (Moscow, Russia, March 24, 2013). – M.: National Research University Higher School of Economics, 2013. – P. 134–148.

325. Pankin, A. V. *Operative Measurements Analyses in Situation Early Recognition Tasks* / A. V. Pankin, A. I. Vodyaho, N. A. Zhukova // Proceedings of the 11th International Conference on Pattern Recognition and Image Analyses: New Information Technologies (Samara, Sept. 23–28, 2013). – Samara: IPSI RAS, 2013. – Vol. 2. – P. 499–452.

326. Программа автоматического многоуровневого синтеза действий интеллектуальных машин [Текст] / В. Ю. Осипов, Н. А. Жукова, Н. В. Климов; патентообладатель Федеральное государственное бюджетное учреждение науки Санкт-Петербургский институт информатики и автоматизации Российской академии наук (СПИИРАН). № 2018664168; заявл. 16.10.2018; опубл. 12.11.2018.

327. Finding motifs in Medical Data / V. Osipov, E. Stankova, A Vodyaho, B. Zeno // 17th International Conference, Computational Science and Its Applications (ICCSA 2017) (Trieste, Italy, July 3–6, 2017). Lecture Notes in Computer Science. – Springer, Cham., 2017. – Part 5. – Vol. 10408. – P. 371–386.

328. Medical Knowledge Representation for Evaluation of Patient’s State Using Complex Indicators / M. Lushnov, V. Kudashov, A. Vodyaho, M. Lapaev, N. Zhukova, D. Korobov // International Conference on Knowledge Engineering and the Semantic Web. – Springer International Publishing, 2016. – С. 344–359.

329. Synthesis of Integral Models of System Dynamics of an Acid-Base State (ABS) of Patients at Operative Measures / D. I. Kurapeev, M. S. Lushnov, V. Yu. Osipov, A. I. Vodyaho, N. A. Zhukova // *Acta Scientific*. – 2019. – Vol. 3, Iss. 3. – P. 16–29.

330. Kurapeev, D. I. Comparison of Integral Models of Systemic Dynamics of Acid-Base State of Venous and Arterial Circulating Blood in Patients with Surgical Interventions / D. I. Kurapeev, M. S. Lushnov, N. A. Zhukova // *Acta Scientific medical sciences*. – 2019. – Vol. 3, Iss. 6. – P. 38–48.

331. Применение полифазных банков фильтров в задачах мониторинга широкого частотного диапазона / А. С. Вознесенский, В. В. Гульванский, Н. А. Жукова, Д. И. Каплун, Д. М. Клионский, А. Л. Олейник, А. А. Петровский // *Известия вузов России. Радиоэлектроника*. – 2013. – Вып. 3. – С. 38–43.

332. Применение алгоритма WOLA в задачах мониторинга широкого частотного диапазона / Д. И. Каплун, Н. А. Жукова, Д. М. Клионский, А. Л. Олейник, А. С. Вознесенский, В. В. Гульванский, А. А. Петровский // *Известия вузов России. Радиоэлектроника*. – 2013. – Вып. 4. – С. 40–45.

333. Витол, А. Д. Разработка редактора онтологий с функцией адаптации к предметной области / А. Д. Витол, Н. А. Жукова // *Науч. сессия МИФИ-2011: Сб. науч. тр.* – М.: МИФИ, 2011. – Т. 3. – С. 62.

334. Nguyen, Ngoc Than. Method for Visualization of Description Logic Formulas / Ngoc Than Nguyen, I. Baymuratov, N. Zhukova // *WSEAS Transactions on Computers*. – 2018. – Vol. 17. – Art. 8. – P. 69–78. – ISSN 1109-2750. – E-ISSN 2224-2872.

335. Баймуратов, И. Р. Метод визуализации онтологий / И. Р. Баймуратов, Н. А. Жукова, Д. И. Муромцев // *Двадцать шестая Междунар. конф. «Математика. Компьютер. Образование»* (г. Пущино, 28 янв. – 2 февр. 2019 г.) [Электронный ресурс]. – Режим доступа: <http://www.mce.su/rus/archive/abstracts/mce26/authors/person283231/doc334698/>

336. Kuvaev, A. Knowledge processing method with calculated functors / A. Kuvaev, V. Meltsov, N. Zhukova // *International Conference Cyber-Physical Systems and Control (CPS&C'2019)* (Saint-Petersburg, June 10–12, 2019).

337. ОАО Zodiac Interactive [Электронный ресурс]. – Режим доступа: <http://www.zodiac.tv/>

338. Когнитивные технологии в управлении мониторингом / В. Ю. Осипов, Н. А. Жукова, М. А. Червонцев, А. И. Водяхо, Н. В. Климов // *Материалы конференции «Информационные технологии в управлении» (ИТУ-2018)* (СПб., 2–4 окт. 2018 г.). – СПб.: АО Концерн «ЦНИИ „Электроприбор”», 2018. – С. 39–48.

339. Программа многоуровневого синтеза процессов когнитивного мониторинга / Н. А. Жукова, С. А. Подкорытов, А. И. Водяхо, А. Б. Тристанов, Н. В. Климов // Известия КГТУ. – 2018. – № 51. – С. 161–174.
340. Klimov, N. Program System for Object Models Deductive Synthesis / N. Klimov, N. Zhukova, N. Andriyanova // Proceedings of the 24th Conference of Open Innovations Association FRUCT (Moscow, Russia, April 08–12, 2019). – 2019. – Art. 116. – P. 803–809.
341. Когнитивные информационные системы мониторинга [Текст]: Монография / А. В. Васильев, А. И. Вайнтрауб, А. И. Водяхо, Н. А. Жукова, Д. И. Курапеев, М. С. Лушнов, Н. А. Смирнов. – СПб.: Изд-во СПбГЭТУ «ЛЭТИ», 2017. – 200 с.
342. Ericsson A. Simple Network Management Protocol (SNMP), 2019 [Электронный ресурс]. – Режим доступа: <http://erlang.org/doc/apps/snmp/snmp.pdf>
343. TR-069 CPE WAN Management Protocol. Version: 1.4 [Электронный ресурс]. – Режим доступа: [https://www.broadband-forum.org/download/TR-069\\_Amendment-6.pdf](https://www.broadband-forum.org/download/TR-069_Amendment-6.pdf)
344. Hazelcast, Inc. [Электронный ресурс]. – Режим доступа: <https://hazelcast.com/>
345. Научно-инженерный центр Санкт-Петербургского электротехнического университета – АО «НИЦ СПб ЭТУ» [Электронный ресурс]. – Режим доступа: <http://rec-etu.com/index.php/ru/>
346. Научно-инженерный центр электротехнического университета (АО «НИЦ ЭТУ») [Электронный ресурс]. – Режим доступа: <http://www.nicetu.ru/>
347. ОКР «Радиент». Система обеспечения надежной и безопасной эксплуатации ракетно-космической техники и наземной космической инфраструктуры космодрома «Плесецк». – 2009–2013.
348. ОКР «Математика-ПИК». Программно-алгоритмическое обеспечение средств полигонных измерительных комплексов. – 2008–2015.
349. ОКР «Интеграция». Распределенная система оперативного координационного планирования применения средств НАКУ КА. – 2010–2013.
350. НИР «Радиент». Разработка научно-методического обеспечения оценивания надежности и безопасности объектов ракетно-космической техники при длительной эксплуатации, обоснования принципов построения и развития автоматизированной системы сбора, анализа и обмена информацией о техническом состоянии и надежности на примере РКК «Союз-2» и КА «Меридиан» космодрома «Плесецк». – 2006–2008.
351. ОКР «Моренос». Разработка средств национального технического контроля испытаний стратегического вооружения в соответствии с международными договорами. – 2006–2010.

352. ОКР «Поиск-СНВ». Разработка аппаратно-программного комплекса исследования характеристик баллистических ракет США по телеметрической информации, предоставляемой в соответствии с Договором о СНВ (АПК ТМИ СНВ). – 2002–2004.

353. ОКР «Адекватность». Разработка АПК исследования характеристик КРН США, АБР и элементов преодоления ПРО ракет США по ТМИ, передаваемой по Договору о СНВ (АПК КРН СНВ). – 2004–2006.

354. ОКР «Ленинградка». Разработка аппаратно-программного комплекса баз данных моделирования и оценивания летно-технических характеристик МБР и БРПЛ США по предоставляемой ТМИ с целью повышения оперативности и качества сравнительного анализа, формирования статистик (построения таксонов) (АПК БД СНВ). – 2005–2007.

355. ОКР «Модернизация». Разработка новых технологий идентификации и обработки телеметрической информации стратегических МБР США в 281 НИЦ ПКИ по данным от национальных технических средств контроля (ПТК ОИИ НТСК). – 2008–2010.

356. ОКР «ИАС-М». Разработка объединенного комплекса обмена информацией между государствами-участниками международных договоров и соглашений. – 2012–2013.

357. Практическая работа «Центр». Дополнение аппаратно-программных средств изделия 14Б873. – 2012–2013.

358. ОКР «Признак». Модернизация аппаратно-программного комплекса контроля параметров летных испытаний межконтинентальных баллистических ракет и баллистических ракет подводных лодок Соединенных Штатов Америки по данным телеметрической информации, поступающей от национальных технических средств контроля и по дипломатическим каналам в рамках Договора о стратегических наступательных вооружениях. – 2013–2015.

359. ОКР «Перспектива-ПИК». Автоматизированная система информационно-аналитического обеспечения испытаний агрегатов, систем и изделий РК СН при пусках из позиционных районов 13 РД. – 2012–2013.

360. ОКР «Диагностика – НИЦ». Разработка материалов в составную часть ОКР «Диагностика» в части методов и алгоритмов обработки и анализа больших массивов многопараметрической информации о состоянии стартового комплекса и ракет космического назначения. – 2010.

361. НИР Листва-2000-НЦ. Интеллектуализация методов решения задач предстартовой подготовки и обработки телеметрической и внешнетраекторной информации применительно к анализу параметров полета объектов РКТ. – № НЦ 05/08 от 11 янв. 2005 г. – 2005.

362. ОКР «Указчик-ВКО/ПТК ОДПП и ОРИ». Создание программно-технического комплекса информационно-аналитического обеспечения деятельности НИИЦ ЦНИИ ВВКО при подготовке, проведении и оценке результатов испытаний МПИК, его основных частей и образцов ВВТ ВКО, испытываемых на МПИК. – 2014.

363. Балтрашевич, В. Э. Интеллектуальная обработка телеметрической информации: Тез. докл. / В. Э. Балтрашевич, Н. А. Жукова, Д. В. Тихонов // Науч. сессия МИФИ-2005: Сб. науч. тр. – М.: МИФИ, 2005. – Т. 3. – С. 124–125.

364. Балтрашевич, В. Э. Интеллектуальная обработка телеметрической информации: Тез. докл. / В. Э. Балтрашевич, Н. А. Жукова // Конф. «Технологии Microsoft в теории и практике программирования» (СПб., 1–2 марта 2005 г.): Сб. науч. тр. – СПб.: Изд-во СПбГПУ, 2005. – С. 165–166.

365. Балтрашевич, В.Э. Интеллектуальная обработка телеметрической информации на основе алгоритмов ассоциации: Тез. докл. / В. Э. Балтрашевич, Н. А. Жукова, Д. В. Тихонов // Науч. сессия МИФИ-2006: Сб. науч. тр. – М.: МИФИ, 2006. – Т. 3. – С. 69–70.

366. Жукова, Н. А. Система обработки телеметрической информации на основе алгоритмов ассоциации: Тез. докл. / Н. А. Жукова, А. В. Мазурин, Д. В. Тихонов // IV регион. конф. «Практика применения научного программного обеспечения в образовании и научных исследованиях» (СПб., 2–3 февр. 2006 г.): Сб. тр. – СПб.: Изд-во Политехн. ун-та, 2006. – С. 25–27.

367. Геппенер, В. В. Система интеллектуальной обработки телеметрической информации / В. В. Геппенер, Н. А. Жукова, Д. В. Тихонов // Междунар. конф. по мягким вычислениям и измерениям (СПб., 27–29 июня 2006 г.): Сб. докл. – СПб.: Изд-во СПбГЭТУ «ЛЭТИ», 2006. – Т. 1. – С.151–155.

368. Жукова, Н. А. Использование алгоритмов ассоциации в интеллектуальных системах обработки телеметрической информации / Н. А. Жукова // Десятая нац. конф. по искусственному интеллекту с междунар. участием (Обнинск, 25–28 сент. 2006 г.): Сб. тр. – М.: Физматлит, 2006. – С.141–149.

369. Жукова Н. А. Обработка телеметрической информации на основе алгоритмов ассоциации / Н. А. Жукова // Известия СПбГЭТУ «ЛЭТИ». Сер. Информатика, управление и компьютерные технологии. – 2006. – Вып. 3. – С. 100–103.

370. Жукова, Н. А. Реализация алгоритмов поиска логических закономерностей в пакете программ интеллектуального анализа данных / Н. А. Жукова, А. В. Мазурин, Д. В. Тихонов // 61-я науч.-техн. конф., посвящ. Дню радио (СПб., апр., 2006 г.): Сб. тр. – СПб.: Изд-во СПбГЭТУ «ЛЭТИ», 2006. – С.18–20.



371. Жукова, Н. А. Использование алгоритмов кластеризации и классификации для обработки телеметрической информации: Тез. докл. / Н. А. Жукова, А. Б. Тристанов // Науч. сессия МИФИ-2007: Сб. науч. тр. – М.: МИФИ, 2007. – Т. 3. – С. 69–70.

372. Геппенер, В. В. Применение секвенциального анализа для обработки телеметрических параметров / В. В. Геппенер, И. В. Горбачева, Н. А. Жукова // Конф. «Технологии Microsoft в теории и практике программирования» (СПб., 13–14 марта 2007 г.): Сб. науч. тр. – СПб.: Изд-во СПбГПУ, 2007. – С. 28.

373. Геппенер, В. В. Система автоматизированной обработки телеметрической информации на основе алгоритмов сегментации и кластер анализа: Тез. докл. / В. В. Геппенер, Н. А. Жукова // Конф. «Технологии Microsoft в теории и практике программирования» (СПб., 13–14 марта 2007 г.): Сб. науч. тр. – СПб.: Изд-во СПбГПУ, 2007. – С. 36.

374. Применение алгоритмов кластеризации и классификации в задачах обработки и интерпретации телеметрической информации / А. В. Васильев, В. В. Геппенер, Н. А. Жукова, Д. М. Клионский, А. Б. Тристанов // Девятая междунар. конф. и выставка «Цифровая обработка сигналов и ее применение» (М., 28–30 марта 2007 г.): Сб. докл. М.: ИПРЖР, 2007. – Вып. IX-2. – С. 389–392.

375. Васильев, А. В. Анализ состояния сложных динамических объектов с использованием алгоритмов интеллектуальной обработки данных: Тез. докл. / А. В. Васильев, В. В. Геппенер, И. В. Горбачева, Н. А. Жукова // Междунар. науч.-техн. конф. «Компьютерное моделирование 2007» (СПб., 26–27 июня 2007 г.): Сб. тр. – СПб.: Изд-во Политехн. ун-та, 2007. – С. 88–90.

376. Васильев, А. В. Методы раннего выявления аномальных ситуаций на основе технологии интеллектуального анализа данных: Тез. докл. / А. В. Васильев, В. В. Геппенер, Н. А. Жукова // 62-ая науч.-техн. конф., посвящ. Дню радио (СПб., апр. 2007 г.): Сб. тр. – СПб.: Изд-во СПбГЭТУ «ЛЭТИ», 2007. – С. 27–29.

377. Васильев, А. В. Методы обработки телеметрической информации на основе алгоритмов Data Mining / А. В. Васильев, В. В. Геппенер, Н. А. Жукова // Известия СПбГЭТУ «ЛЭТИ». Сер. Информатика, управление и компьютерные технологии. – 2007. – Спец. вып. 1. – С. 3–10.

378. Методы сегментации медленно меняющихся телеметрических сигналов / А. В. Васильев, В. В. Геппенер, Н. А. Жукова, Д. М. Клионский // Междунар. конф. по мягким вычислениям и измерениям (СПб., 25–27 июня 2007 г.): Сб. докл. – СПб.: Изд-во СПбГЭТУ «ЛЭТИ», 2007. – Т. 1. – С. 112–115.

379. Жукова, Н. А. Система контроля состояния сложных динамических объектов / Н. А. Жукова // Междунар. конф. по мягким вычислениям и измерениям (СПб., 25–27 июня 2007 г.): Сб. докл. – СПб.: Изд-во СПбГЭТУ «ЛЭТИ», 2007. – Т. 2. – С.80–83.

380. Геппенер, В. В. Обработка телеметрической информации на основе алгоритмов сегментации и секвенциального анализа / В. В. Геппенер, Н. А. Жукова // Известия СПбГЭТУ «ЛЭТИ». Сер. Информатика, управление и компьютерные технологии. – 2007. – Вып. 2. – С. 37–41.

381. Методы обработки телеметрической информации на основе алгоритмов секвенциального анализа / А. В. Васильев, В. В. Геппенер, И. В. Горбачева, Н. А. Жукова, А. Б. Тристанов // Известия СПбГЭТУ «ЛЭТИ». Сер. Информатика, управление и компьютерные технологии. – 2007. – Спец. вып. 2. – С. 24–31.

382. Жукова, Н. А. Методы контроля состояния сложных динамических объектов / Н. А. Жукова, А. Б. Тристанов // Вестник компьютерных и информационных технологий. – 2007. – № 9. – С. 2–10.

383. Vasiljev, A. V. Automatic control system of complex dynamic objects state on the base of telemetering information analysis / A. V. Vasiljev, V. V. Geppener, A. B. Tristanov, A. V. Ecalo, N. A. Zhukova // Proceedings of 8th International Conference on Pattern Recognition and Image Analysis: New Information Technologies (Yoshkar-Ola, Mari State Technical University, Oct. 8–12, 2007). – 2007. – Vol. 2. – P. 376–379.

384. Геппенер, В. В. Методы сокращения признакового пространства в системе анализа телеметрической информации / В. В. Геппенер, Н. А. Жукова, А. Б. Тристанов // Науч. сессия МИФИ-2008: Сб. науч. тр. – М.: МИФИ, 2008. – Т. 10. – С. 170–171.

385. Геппенер, В. В. Система оценки состояния сложных динамических объектов, основанная на использовании результатов совместной обработки накопленных сигнальных данных / В. В. Геппенер, И. В. Горбачева, Н. А. Жукова // Конф. «Технологии Microsoft в теории и практике программирования» (СПб., 11–12 марта 2008 г.): Сб. науч. тр. – СПб.: Изд-во СПбГПУ, 2008. – С. 124.

386. Геппенер, В. В. Разработка онтологии верхнего уровня для организации процесса обработки телеметрической информации / В. В. Геппенер, А. О. Дерипаска, Н. А. Жукова // Конф. «Технологии Microsoft в теории и практике программирования». (СПб., 11–12 марта 2008 г.): Сб. науч. тр. – СПб.: Изд-во СПбГПУ, 2008. – С. 129.

387. Жукова, Н. А. Многомерный секвенциальный анализ телеметрических данных на основе поиска часто встречающихся наборов параметров / Н. А. Жукова, Д. М. Клионский // Десятая межд. конф. и выставка «Цифровая обработка сигналов и ее применение» (М., 26–28 марта 2008 г.): Сб. докл. – М.: Изд-во ИПУ, 2008. – Вып. X-1. – С. 197–200.

388. Организация процесса обработки телеметрической информации с использованием технологии семантических веб-служб / А. В. Васильев, В. В. Геппенер, А. О. Дерипаска, Н. А. Жукова, А. Б. Тристанов // Десятая междунар. конф. и выставка «Цифровая обработка сигналов и ее применение» (М., 26–28 марта 2008 г.): Сб. докл. – М.: Изд-во ИПУ, 2008. – Вып. X-2. – С. 551–554.

389. Метод построения моделей сложных нестационарных процессов на основе кусочно-постоянной аппроксимации / А. В. Васильев, В. В. Геппенер, И. В. Горбачева, Н. А. Жукова, А. Б. Тристанов // Междунар. конф. по мягким вычислениям и измерениям (СПб., 23–25 июня 2008 г.): Сб. докл. – СПб.: Изд-во СПбГЭТУ «ЛЭТИ», 2008. – Т. 1. – С. 164–168.

390. Онтологический подход к обработке телеметрической информации / А. В. Васильев, А. О. Дерипаска, И. В. Горбачева, Н. А. Жукова // Одиннадцатая нац. конф. по искусственному интеллекту с междунар. участием (Дубна, 28 сент. – 3 окт. 2008 г.): Сб. тр. – М.: ЛЕНАНД, 2008. – Т. 2. – С. 139–147.

391. Балтрашевич, В. Э. Система идентификации телеметрических параметров на основе технологии Web-сервисов / В. Э. Балтрашевич, И. В. Ильина, Н. А. Жукова // Системы управления и информационные технологии. – 2008. – № 3 (33). – С. 30–34.

392. Система идентификации телеметрических параметров на основе алгоритмов интеллектуального анализа данных / В. Э. Балтрашевич, А. В. Васильев, Н. А. Жукова, И. В. Ильина // Известия СПбГЭТУ «ЛЭТИ». Сер. Информатика, управление и компьютерные технологии. – 2008. – Вып. 9. – С. 11–16.

393. Горбачева, И. В. Архитектура данных системы интеллектуальной обработки измерительной информации / И. В. Горбачева, А. О. Дерипаска, Н. А. Жукова // Науч. сессия МИФИ-2009: Сб. науч. тр. – М.: МИФИ, 2009. – Т. 3. – С. 89.

394. Оценка качества измерительной информации с использованием методов интеллектуального анализа данных / А. В. Васильев, И. В. Горбачева, А. О. Дерипаска, Н. А. Жукова // Одиннадцатая междунар. конф. и выставка «Цифровая обработка сигналов и ее применение» (М., 25–27 марта 2009 г.): Сб. докл. – М.: Рос. НТОРЭС им. А. С. Попова, 2009. – Вып. XI-2. – С. 540–543.

395. Идентификация сложных динамических объектов с использованием шаблонов телеметрических параметров / А. В. Васильев, В. В. Геппенер, И. В. Горбачева, Н. А. Жукова, А. Б. Тристанов // Нейрокомпьютеры: разработка применение. – 2009. – № 3. – С. 44–49.

396. Идентификация телеметрических параметров с использованием нейронных сетей / А. В. Васильев, В. В. Геппенер, И. В. Горбачева, Н. А. Жукова, И. С. Соколов // Нейрокомпьютеры: разработка, применение. – 2009. – № 11. – С. 39–44.

397. Витол, А. Д. Применение алгоритмов классификации в задаче контроля состояния сложных динамических объектов / А. Д. Витол, А. О. Дерипаска, Н. А. Жукова // Науч. сессия МИФИ-2010: Сб. науч. тр. – М.: МИФИ, 2010. – Т. 3. – С. 63–64.

398. Жукова, Н. А. Разработка подсистемы расчета зон видимости объекта измерительными средствами в среде ГИС / Н. А. Жукова, А. О. Менько // Конф. «Технологии Microsoft в теории и практике программирования» (СПб., 16–17 марта 2010 г.): Сб. науч. тр. – СПб.: Изд-во СПбГПУ, 2010. – С. 49.

399. Жукова, Н. А. Программный комплекс обработки оперативной информации от сложных динамических объектов / Н. А. Жукова, А. Р. Лисс // Известия СПбГЭТУ «ЛЭТИ». Сер. Информатика, управление и компьютерные технологии. – 2010. – Вып. 5. – С. 67–72.

400. Интеллектуальный подход к анализу структуры и семантики ГТС / В. Э. Балтрашевич, А. Д. Витол, Н. А. Жукова, И. С. Соколов // Двенадцатая нац. конф. по искусственному интеллекту с междунар. участием (Тверь, 20–24 сент. 2010 г.): Сб. тр. – М.: Физматлит, 2010. – Т. 2. – С. 47–56.

401. Менько, А. О. Интеллектуальная система планирования проведения сеансов измерений / А. О. Менько, Н. А. Жукова // Науч. сессия МИФИ-2011: Сб. науч. тр. – М.: МИФИ, 2011. – Т. 3. – С. 67–68.

402. Жукова, Н. А. Организация процессов адаптивной обработки и анализа телеметрической информации / Н. А. Жукова // Всерос. молодежн. науч.-техн. конф. «Прикладные научно-технические проблемы современной теории управления системами и процессами» (М., 24 окт. 2012 г.): Сб. тез. докл. – М.: Концерн «Вега», 2012. – С. 6.

403. Васильев, А. В. Методы оценивания информативности результатов опытной эксплуатации сложных многопараметрических объектов в условиях неопределенности / А. В. Васильев, Н. А. Жукова // Известия СПбГЭТУ «ЛЭТИ». Сер. Информатика, управление и компьютерные технологии. – 2013. – Вып. 9. – С. 63–67.

404. Вайнтрауб, А. И. Применение интеллектуальных методов анализа на примере обработки информации, получаемой с ракет-носителей типа «Союз» в процессе пуска / А. И. Вайнтрауб, С. А. Беляев, Н. А. Жукова // III Всерос. науч.-техн. конф. «Теоретические и прикладные проблемы развития и совершенствования автоматизированных систем управления военного назначения» (СПб., 22 нояб. 2017 г.): Сб. докл. – СПб.: Военно-космическая акад. им. А. Ф. Можайского, 2017. – С. 202–203.

405. Вайнтрауб, А. И. Применение интеллектуальных методов анализа на примере обработки информации, получаемой с ракет-носителей типа «Союз» в процессе пуска / А. И. Вайнтрауб, С. А. Беляев, Н. А. Жукова // Авиационная и ракетно-космическая техника. – 2018. – № 4. – С. 11–25.

406. Витол, А. Д. Метод построения статистических моделей технологических циклов управления космическими аппаратами / А. Д. Витол // Известия СПбГЭТУ «ЛЭТИ». Сер. Информатика, управление и компьютерные технологии. – 2012. – Вып. 7. – С. 31–37.

407. IRIG 106 Document 106-15. Part II: TELEMETRY STANDARDS [Электронный ресурс]. – Режим доступа: <https://ru.scribd.com/document/387174067/IRIG-106-15-Telemetry-Standards>

408. Северо-Западный федеральный медицинский исследовательский центр им. В. А. Алмазова (ФГБУ «СЗФМИЦ им. В. А. Алмазова» Минздрава России) [Электронный ресурс]. – Режим доступа: <http://www.almazovcentre.ru/>

409. Medical text processing for SMDA project / M. Lushnov, T. Sanfin, M. Lapaev, N. Zhukova // ESWC 2016 (Heraklion, Greece, May 29, 2016) // RichMedSem Workshop CEUR Proceedings. – 2016. – Vol. 1613. – P. 1–13.

410. Система обработки текстовых медицинских данных / М. В. Лапаев, А. И. Водяхо, А. Б. Смирнов, Н. А. Жукова // Известия СПбГЭТУ «ЛЭТИ». – 2016. – № 9. – С. 16–21.

411. Vodyaho, A. Semantic medical care in smart cities / A. Vodyaho, N. Zhukova, A. Koltavskiy // REAL CORP 2016 – SMART ME UP! How to become and how to stay a Smart City, and does this improve quality of life. Proceedings of 21st International Conference on Urban Planning, Regional Development and Information Society. – 2016. – P. 551–559 [Электронный ресурс]. – Режим доступа: <http://repository.corp.at/168/>

412. Модели безопасности в управлении процессами медицинской деятельности в семантических информационных системах / М. Лапаев, Н. Жукова, Р. Хайдарова, М. Лушнов // Материалы 5-й Рос. мультikonф. по проблемам управления «Информационные технологии в управлении» (СПб., 9–11 окт. 2016 г.). – СПб.: Концерн «ЦНИИ „Электроприбор”», 2016.

413. Zhukova, N. A. Extraction of named entities from semi-structured texts for medical domain. / N. A. Zhukova, M. I. Berezov, S. Levedev, E. Zavadskaya // 6th International Conference on Analysis of Images, Social Networks, and Texts (AIST'2017) // CEUR Workshop Proceedings. – 2017. – Vol. 1975. – P. 31–40 [Электронный ресурс]. – Режим доступа: <http://ceur-ws.org/>

414. An Ontology-driven Toolset for Fast Prototyping of Medical Data Processing Systems / S. Lebedev, A. Vodyaho, N. Zhukova, D. Kurapeev, M. Lushnov // International Journal of Biology and Biomedical Engineering. – 2017. – Vol. 11. – P. 135–142.

415. Куперштох, В. Л. Сумма внутренних связей как показатель качества классификации / В. Л. Куперштох, Б. Г. Миркин, В. А. Трофимов // Автоматика и телемеханика. – 1976. – № 3. – С. 133–141; Automation and Remote Control. – 1976. – Vol. 37 (3). – P. 433–440.

416. Боярский, К. К. Семантико-синтаксический парсер SemSin / К. К. Боярский, Е. А. Каневский // Научно-технический вестник информационных технологий, механики и оптики. – 2015. – Т. 15. – № 5. – С. 869–876.

417. Международная классификация болезней 10-го пересмотра (МКБ-10) [Электронный ресурс]. – Режим доступа: <http://mkb-10.com/>

418. Научно-исследовательская лаборатория объектно-ориентированных геоинформационных систем СПИИРАН [Электронный ресурс]. – Режим доступа: <http://www.spiiras.nw.ru>

419. ОКР «Алеврит». Создание программно-аппаратного комплекса автоматизации процесса управления функциональной системой освещения обстановки в составе интегрированной АСУ пространственными процессами на море. – 2003–2007.

420. ОКР «Автоматизм». Оперативно-тактический тренажерный комплекс Военно-Морского Флот. – 2008–2012.

421. СППР «Автоматизм». Разработка исполнений ОТТК для обеспечения поддержки принятия решения ОВУ. – 2012.

422. ОКР «Галтель-Алеврит». Разработка автоматизированной геоинформационной системы для решения задач мониторинга дна моря в пунктах базирования сил ВМФ. – 2010–2011.

423. Дерипаска, А. О. Организация обработки многомерных измерений на основе технологии ИГИС / А. О. Дерипаска, Н. А. Жукова, А. В. Панькин // Материалы Четвертой Всерос. науч.-техн. конф. «Технические проблемы освоения мирового океана» (Владивосток, 3–7 окт. 2011 г.). – Владивосток: Изд-во «Дальнаука» ДВО РАН, 2011. С. 272–278.

424. Жукова, Н. А. Теоретические основы построения систем адаптивной обработки и анализа многомерных измерений на базе интеллектуальных геоинформационных систем / Н. А. Жукова // Материалы XIII Санкт-Петербургской междунар. конф. «Региональная информатика» (СПб., 24–26 окт. 2012 г.). – СПб.: СПОИСУ, 2012. – С. 366.

425. Жукова, Н. А. Метод адаптивной верификации измерений параметров водной среды на основе технологий интеллектуальных геоинформационных систем / Н. А. Жукова // Материалы Пятой Всерос. науч.-техн. конф. «Технические проблемы освоения мирового океана» (Владивосток, 30 сент. – 4 окт. 2013г.). – Владивосток: Изд-во «Дальнаука» ДВО РАН, 2013. – С.445–450.

426. Жукова, Н. А. Организация систем адаптивной обработки и анализа многомерных измерений параметров пространственно соотнесенных объектов на основе интеллектуальных ГИС / Н. А. Жукова // II Междунар. науч.-практ. конф. «Актуальные

проблемы современной науки» (Ставрополь, п. Домбай, 13–16 марта 2013 г.): Сб. тр. – Ставрополь: Изд-во СевКавГТУ, 2013.

427. Zhukova, N. A. Method for adaptive multidimensional measurements processing based on IGIS technologies / N. A. Zhukova // Proceedings of the 11th International Conference on Pattern Recognition and Image Analyses: New Information Technologies (Samara, Sept. 23–28, 2013). – Samara: IPSIRAS, 2013. – Vol. 1. – P. 366–369.

428. Жукова, Н. А. Архитектурный фреймворк для разработки адаптивных интеллектуальных геоинформационных систем: Тез. докл. / Н. А. Жукова, А. И. Водяхо // Науч. сессия МИФИ-2014: Сб. науч. тр. – М.: МИФИ, 2014. – Т. 3. – С. 15.

429. Жукова, Н. А. Шаблоны проектирования объектно-ориентированных интеллектуальных геоинформационных систем / Н. А. Жукова // Известия СПбГЭТУ «ЛЭТИ». Сер. Информатика, управление и компьютерные технологии. – 2014. – Вып. 6. – С. 20–26.

430. Жукова, Н. А. Интеллектуальные геоинформационные технологии для управления комплексами технических объектов / Н. А. Жукова // Материалы 7-й Рос. мультikonф. по проблемам управления «Информационные технологии в управлении» (СПб., 7–9 окт. 2014 г.). – СПб.: Концерн «ЦНИИ „Электроприбор“», 2014. – С. 280–289.

431. Smirnova, O. V. Atmosphere and ocean data processing in decision making support system for Arctic exploration / O. V. Smirnova, N. A. Zhukova // Proceedings of the 6th International Workshop on Information Fusion and Geographic Information Systems: Environmental and Urban Challenges (Saint-Petersburg, May 12–15, 2013). – Berlin; Heidelberg: Springer, 2014. – P. 305–324.

432. Ignatov, D. I. Dynamic Information Model for Oceanographic Data Representation / D. I. Ignatov, O. V. Smirnova, N. A. Zhukova // Proceedings of the Workshop «Modeling States, Events, Processes and Scenarios» co-located with 20th International Conference on Conceptual Structures (India, Mumbai, Jan. 10–12, 2013). – М.: Higher School of Economics Publishing House, 2013. – P. 82–97.

433. Жукова, Н. А. Архитектура данных системы поддержки принятия решений о состоянии водного пространства Арктического региона: Тез. докл. / Н. А. Жукова // Науч. сессия МИФИ-2013: Сб. науч. тр. – М.: МИФИ, 2013. – Т. 2. – С. 24.

434. Climate Data Analysis by IGIS / F. R. Galyano, V. V. Popovich, M. Schrenk, N. A. Zhukova // Proceedings of 18th International Conference on Urban Planning and Regional Development in the Information Society (Italy, Rome, May 20–23, 2013). – Schwechat: CORP, 2013. – P. 171–179.

435. Жукова, Н. А. Методы адаптивной обработки и анализа метеорологических и океанографических данных в Арктическом регионе / Н. А. Жукова, О. В. Смирнова // Труды СПИИРАН. – 2013. – № 7. – С. 258–272.

436. Арктический и антарктический научно-исследовательский институт [Электронный ресурс]. – Режим доступа: <http://www.aari.ru/>

437. НИР ONR-Global № 62909-12-1-7013. Decision Making Support System for Arctic Exploration, Monitoring and Governance = Система поддержки принятия решений по мониторингу состояния Арктического региона / СПИИРАН. – 2012.

438. Жукова, Н. А. Интеллектуальная геоинформационная система мониторинга дна моря / Н. А. Жукова, Я. А. Ивакин, А. В. Панькин // Материалы Четвертой Всерос. науч.-техн. конф. «Технические проблемы освоения мирового океана» (Владивосток, 3–7 окт. 2011 г.). – Владивосток: Изд-во «Дальнаука» ДВО РАН, 2011. – С. 20–25.

439. Жукова, Н. А. Необитаемые подводные аппараты как компонент системы поддержки принятия решения / Н. А. Жукова, А. В. Панькин // Материалы Пятой Всерос. науч.-техн. конф. «Технические проблемы освоения мирового океана» (Владивосток, 30 сент. – 4 окт. 2013 г.). – Владивосток: Изд-во «Дальнаука» ДВО РАН, 2013. – С. 435–440.

440. Институт проблем морских технологий ДВО РАН [Электронный ресурс]. – Режим доступа: <http://www.imtp.febras.ru/>

441. Сеть ныряющих буев Арго [Электронный ресурс]. – Режим доступа: <http://www.argo.ucsd.edu>

442. Центры данных буев Арго [Электронный ресурс]. – Режим доступа: <http://www.argodatamgt.org/>; <ftp://ftp.ifremer.fr/ifremer/argo>

443. Argos User's Manual [Электронный ресурс]. – Режим доступа: <http://www.argos-system.org/manual/>

444. World Ocean Atlas 2009 / R. A. Locarnini, A. V. Mishonov, J. I. Antonov, T. P. Boyer, H. E. Garcia, O. K. Baranova, M. M. Zweng, D. R. Johnson. – Vol. 1: Temperature. [Электронный ресурс]. – Режим доступа: [https://www.researchgate.net/publication/216028010\\_World\\_Ocean\\_Atlas\\_2009\\_Vol\\_1\\_Temperature](https://www.researchgate.net/publication/216028010_World_Ocean_Atlas_2009_Vol_1_Temperature)

445. World Ocean Atlas 2009 / J. I. Antonov, D. Seidov, T. P. Boyer, R. A. Locarnini, A. V. Mishonov, H. E. Garcia, O. K. Baranova, M. M. Zweng, D. R. Johnson. – Vol. 2: Salinity [Электронный ресурс]. – Режим доступа: <https://www.nodc.noaa.gov/OC5/WOA09/pubwoa09.html>

446. Objective Analyses of Annual, Seasonal, and Monthly Temperature and Salinity for the World Ocean on a 1/4 degree Grid / T. P. Boyer, S. Levitus, H. E. Garcia, R. A. Locarnini, C. Stephens, J. I. Antonov // International Journal of Climatology. – 2004. – Vol. 25. – P. 931–945.



447. Кораблев, А. А. Создание океанографической базы данных для мониторинга климата в Северо-Европейском бассейне Арктики / А. А. Кораблев, А. В. Пнюшков, А. В. Смирнов // Труды ААНИИ. – 2007. – Т. 447. – С. 85–108.

448. Xie, J. Ensemble optimal interpolation schemes for assimilating Argo profiles into a hybrid coordinate ocean model / J. Xie, J. Zhu // Ocean Modelling. – 2010. – Vol.33, Iss. 3–4. – P. 283-298.

449. Four-dimensional variational ocean reanalysis: a 30-year high-resolution dataset in the western North Pacific (FORA-WNP30) / N. Usui, T. Wakamatsu, Y. Tanaka, N. Hirose, T. Toyoda, S. Nishikawa, Y. Fujii, Y. Takatsuki, H. Igarashi, H. Nishikawa, Y. Ishikawa, T. Kuragano, M. Kamachi // Journal of Oceanography. – 2017. – Vol. 73, Iss. 2. – P. 205–233.

450. Ensemble Kalman methods for high-dimensional hierarchical dynamic space-time models / M. Katzfuss, J. R. Stroud, C. K. Wikle [Электронный ресурс]. – Режим доступа: <https://arxiv.org/pdf/1704.06988.pdf>

451. Разработка программных средств поддержки малогабаритного волоконно-оптического устройства БИНС-500К. – 2010–2011. – № 6937/МО-79.

452. Внесение изменений в КПС «Декларант ЭТДиТС» и входящие в состав АПС «Электронное представление сведений» ПЗ «Администрирование – ДКЛ», КПС «Эксперт – ДКЛ», ПЗ «Мониторинг – ДКЛ» и КПС «Электронный архив декларанта». – 2010. – № 13-71/МО-87.

453. Организация высокотехнологичного производства систем гидроакустического мониторинга акватории на базе покровных антенн в местах размещения нефте- и газодобывающих платформ в районе Арктического шельфа. – 2013–2014. – № 13. – G25.31.0054/ФКТИ I. – Договор от «22» окт. 2010 г. с Минобрнауки России.

454. Создание высокопроизводительных вычислительных технологий для интеллектуальных систем оперативной обработки и визуализации гидроакустической информации – 2012-1.1-12-000-2010-011. – Грант в форме субсидии из федерального бюджета для финансового обеспечения расходов, связанных с выполнением научно-исследовательской работы. – 2012–2013.

455. Мониторинг широкого частотного диапазона с использованием банков цифровых фильтров на высокопроизводительной аппаратно-программной базе. – 2012-1.5-12-000-2010-002. – Грант в форме субсидии из федерального бюджета для финансового обеспечения расходов, связанных с выполнением научно-исследовательской работы. – 2012.

456. НИОКР «Разработка системы контроля состояния сложных динамических объектов» / СПб ГЭТУ «ЛЭТИ». – 2007-288. – № У-2007-3/1-3.

457. Разработка программных модулей для тренажера по обучению боевого расчета проверке состояния 3 ступени РН «Союз-2». – 2010. – № 6947/УНИЦ ПСКС-7.

458. НИР «Обработка оперативных измерений параметров водной среды с использованием возможностей интеллектуальных геоинформационных систем». Программа фундаментальных исследований Президиума РАН № 23 «Фундаментальные проблемы океанологии: физика, геология, биология, экология». – 2012–2013.

459. НИР «Прогнозные исследования в области проблем развития когнитивной геопространственной информационной поддержки процессов принятия решений для систем управления перспективных робототехнических комплексов вооружения различного базирования» (шифр «Эстафета-Ф-СПИИРАН»). – 2013.

460. Модуль построения предикторов значений параметров систем организма / Д. И. Муромцев, Н. А. Жукова, С. В. Лебедев; патентообладатель Федеральное государственное автономное образовательное учреждение высшего образования «Санкт-Петербургский национальный исследовательский университет информационных технологий, механики и оптики». № 2018612644; заявл. 29.12.2017; опубл. 21.02.2018.

461. City Computer Architecture. Corporative Information Systems Approach / V. V. Popovich, M. Shrenk, O. V. Smirnova, Y. U. Shalamaiko, N. A. Zhukova // Proceedings of 16th International Conference on Urban Planning, Regional Development and Information Society (Germany, Essen, 18–20 May 2011). – Schwechat: CORP, 2011. – P.105–111.

462. Kuriashkin, V. L. Data analysis methods for urban planning. Problem-oriented stakeholders maps building / V. L. Kuriashkin, N. A. Zhukova // Proceedings of 18th International Conference on Urban Planning and Regional Development in the Information Society (Italy, Rome, May 20–23, 2013). – Schwechat: CORP, 2013. – P. 233–237.

463. Pankin, A. V. Data Representation Dynamic Model for Distributed Urban IGIS / A. V. Pankin, N. A. Zhukova // Proceedings of 18th International Conference on Urban Planning and Regional Development in the Information Society (Italy, Rome, May 20–23, 2013). – Schwechat: CORP, 2013. – P. 1139–1145.

464. Galiano, F. R. Scenario Approach for Image Processing in Smart City / F. R. Galiano, M. Pelevin, N. Zhukova // Proceedings of 19th International Conference on Urban Planning and Regional Development in the Information Society (Austria, Vienna, May 21–23, 2014). – Vienna: CORP, 2014. – P. 497–505.

465. Vodyaho, A. Building Smart Applications for Smart Cities – IGIS-based Architectural Framework / A. Vodyaho, N. Zhukova // Proceedings of 19th International Conference on Urban Planning and Regional Development in the Information Society (Austria, Vienna, May 21–23, 2014). – Schwechat: CORP, 2014. – P. 109–118.

466. Zhukova, N. A. Technological Solutions for Knowledge Management in Smart Cities / N. A. Zhukova // Proceedings of 19th International Conference on Urban Planning and Regional Development in the Information Society (Austria, Vienna, May 21–23, 2014). – Vienna: CORP, 2014. – P. 653–664.

467. Smirnova, O. Smart Navigation for Modern Cities / O. Smirnova, N. Zhukova // Proceedings of 19th International Conference on Urban Planning and Regional Development in the Information Society (Austria, Vienna, May 21–23, 2014). – Vienna: CORP, 2014. – P. 593–602.

468. Vodyaho, A. I. Data Science Technologies for Vibrant Cities / A. I. Vodyaho, N. A. Zhukova, O. Smirnova // REAL CORP 2015. PLAN TOGETHER – RIGHT NOW – OVERALL. From Vision to Reality for Vibrant Cities and Regions. Proceedings of 20th International Conference on Urban Planning, Regional Development and Information Society (Ghent, Belgium, May 5–7, 2015.). – P. 209–218 [Электронный ресурс]. – Режим доступа: <https://repository.corp.at/4/>

469. Автоматизированное формирование профилей объектов научно-образовательной деятельности в информационной системе управления университета / Д. А. Вареников, Н. А. Жукова, Д. И. Муромцев, М. Д. Шлей, М.Н. Ефимов // Известия высших учебных заведений. Приборостроение. – 2015. – Т. 58, № 12. – С. 1022–1026.

470. Система информационной поддержки научной составляющей образовательного процесса / Д. А. Вареников, М. Д. Шлей, А. И. Водяхо, Н. А. Жукова // Известия СПбГЭТУ «ЛЭТИ». – 2016. – № 6. – С. 16–21.

471. Navrotskiy, M. Scientific portal of university department-shaping research area of users through their behavior / M. Navrotskiy, N. Zhukova // CEUR Workshop Proceedings. – 2017. – Vol. 1710. – P. 101–110.

472. Навроцкий, М. А. Онтология проектирования, применения и сопровождения порталов научно-технической информации / М. А. Навроцкий, Н. А. Жукова, Д. И. Муромцев // Онтология проектирования. – 2018. – Т. 8. – № 1 (27). – С. 96–109.

473. Методология проектирования, разработки и сопровождения доменных семантических порталов научно-технической информации / М. А. Навроцкий, Н. А. Жукова, Д. И. Муромцев, Н. Г. Мустафин // Научно-технический вестник информационных технологий, механики и оптики. – 2018. – Т. 18. – № 2. – С. 286–298.

474. Архитектурные решения для порталов научно-технической информации / М. А. Навроцкий, Д. И. Муромцев, Н. А. Жукова, А. И. Водяхо // Известия СПбГЭТУ «ЛЭТИ». – 2018. – № 5. – С. 28–39.

475. Безопасное взаимодействие человека с интернет пространством / В. Ю. Осипов, В. В. Никифоров, М. С. Лушнов, М. А. Щукина, Н. А. Жукова // Материалы 4-й Междунар.

науч. конф. «Технологическая перспектива в рамках Евразийского пространства: новые рынки и точки экономического роста» (СПб., 13–15 дек. 2018 г.) / Под. ред. М. И. Барабановой и др. – СПб.: Астерион, 2018. – С. 180–188.

476. Development of an Intelligent Module for Monitoring and Analysis of Client's Bank Transactions / V. Meltsov, P. Novokshonov, D. Repkin, A. Kuvaev, N. Zhukova // Proceedings of the 24th Conference of Open Innovations Association FRUCT (Moscow, Russia, April 08–12, 2019). – 2019. – Art. 36. – P. 255–262.

477. Методы кластеризации в обработке геофизических данных / В. В. Геппенер, Н. А. Жукова, А. Б. Тристанов, П. П. Фирстов // Междунар. конф. по мягким вычислениям и измерениям (СПб., 25–27 июня 2007 г.): Сб. докл. – СПб.: Изд-во СПбГЭТУ «ЛЭТИ», 2007. – Т. 2. – С. 186–188.

478. Методы и модели интеллектуального анализа сигналов геофизических полей / Н. А. Жукова, В. В. Геппенер, А. Б. Тристанов, А. В. Экало // Нейрокомпьютеры: разработка и применение. – 2007. – № 6. – С. 49–54.

479. Жукова, Н. А. Архитектура системы обработки и анализа комплекса геофизических данных / Н. А. Жукова, А. Б. Тристанов, П. П. Фирстов // Материалы Всерос. конф. «Современные информационные технологии для научных исследований» (Магадан, 20–24 апр. 2008 г.). – Магадан: СВНЦ ДВО РАН, 2008. – С. 192–194.

480. Тристанов, А. Б. Сегментация связанных временных рядов на примере объемной активности радона и атмосферного давления / А. Б. Тристанов, П. П. Фирстов, Н. А. Жукова // Вестник КРАУНЦ. Физ.-мат. науки. – 2010. – № 1 (1). – С. 62–67.

481. Жукова, Н. А. Исследование метода на основе использования смешанных гауссовских моделей в задаче оценки вариативности диктора / Н. А. Жукова, К. К. Симончик // Одиннадцатая междунар. конф. и выставка «Цифровая обработка сигналов и ее применение» (М., 25–27 марта 2009 г.): Сб. докл. – М.: Рос. НТОРЭС им. А. С. Попова, 2009. – Вып. XI-1. – С. 271–274.

482. Zamula, D. Mobile Museum Guides Applications based on Knowledge Graphs / D. Zamula, D. Muromtsev, N. Zhukova // Proceedings of 22nd International Conference on Urban Planning, Regional Development and Information Society (Vienna, Austria, Sept. 12–14, 2017). – Vienna: CORP, 2017. – P. 365–372.

483. Программа автоматического многоуровневого синтеза моделей объектов мониторинга [Текст] / В. Ю. Осипов, Н. А. Жукова, Н. В. Климов; патентообладатель Федеральное государственное бюджетное учреждение науки Санкт-Петербургский институт информатики и автоматизации Российской академии наук (СПИИРАН). № 2019662386; заявл. 10.10.2019; опубл. 14.10.2019.

484. Жукова Н.А. Общая и частные задачи многоуровневого синтеза моделей объектов мониторинга // Научно-техническая информация. Сер. 2, Информационные процессы и системы. – 2019. – № 11. – С. 16–22.

## Приложение 1. Результаты построения моделей объектов по данным мониторинга на примере области телекоммуникаций

### П.1. Постановка задачи построения моделей объектов по данным мониторинга в области телекоммуникаций

В области телекоммуникаций высоко востребованы системы мониторинга и управления распределенными техническими объектами. Это определяется особенностями среды, в которой реализуются эти системы, в частности, распределенностью и структурной сложностью сетей операторов, высокими требованиями, предъявляемыми к надежности работы таких сетей. Неустойчивая работа сети приводит к потере абонентов. Особенное внимание проблемам мониторинга состояния сетей и их управления уделяется операторами кабельного телевидения. В настоящее время кабельное телевидение достаточно широко распространено, особенно в США и Канаде.

Современные сети кабельного телевидения насчитывают сотни тысяч и даже миллионы абонентов, причем число их продолжает расти. Сети представляют собой распределенные системы, в состав которых входят сети передачи данных, серверное и абонентское оборудование. Типовая структура сети кабельного цифрового телевидения (СКЦТ) показана на рисунке П.1.

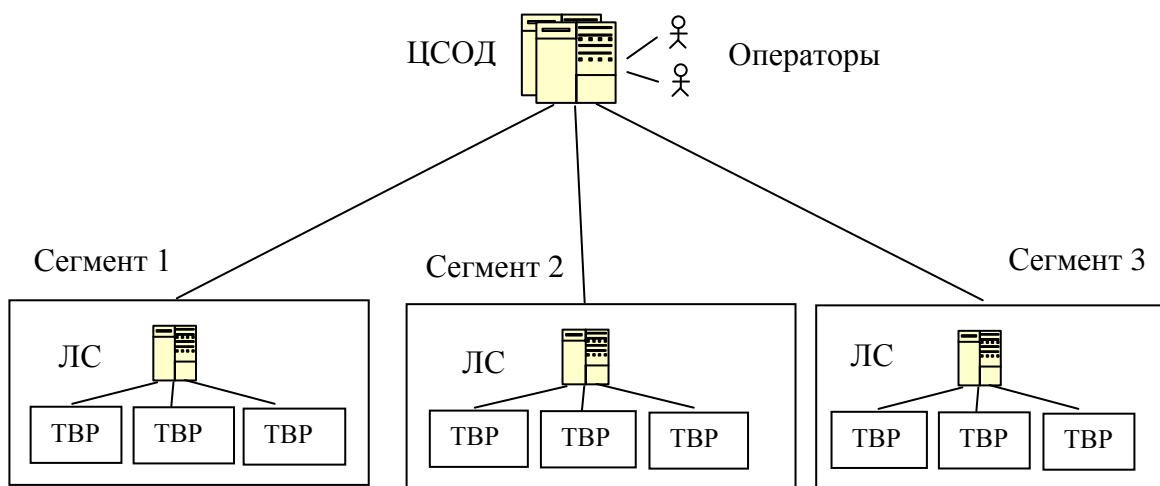


Рисунок П.1 – Типовая структура СКЦТ

На рисунке П.1 используются следующие обозначения: ЦСОД- центр сбора и обработки данных операторов СКЦТ; ТВР - ТВ-ресиверы (далее по тексту может использоваться термин «ресивер»), которые обеспечивают базовую и расширенную функциональность телевизоров конечным пользователям; ЛС – локальная сеть. Один центр сбора и обработки данных оператора сети может обслуживать несколько независимых сегментов сети через выделенные локальные сервера, разделяющие устройства по территориальному и иным признакам. В дальнейшем для обозначения серверной и

клиентской составляющей также используются англоязычные наименования и их сокращения, принятые в предметной области, в частности, Advanced Messaging Solution (AMS), Set-top-box (STB).

СКЦТ имеют следующие особенности:

- большой и очень большой размер сетей,
- высокая динамичность среды,
- гетерогенность сетевой инфраструктуры,
- жесткие требования в плане совокупной стоимости владения, надежности, скорости реакции и другие.

Отказ даже отдельных элементов сети может привести к возникновению эффектов “лавин”, когда в результате локальной неполадки возникает лавинообразное увеличение трафика между локальным сервером и ТВ-ресиверами из-за многократных попыток получения / передачи данных.

Значительная часть ошибок возникает на самих ТВР, а также при их взаимодействии с сетью. Возможности по своевременному выявлению ошибок, их локализации и устранению существенно ограничены. Основные ограничения, возникающие при сборе данных, определяются низкой пропускной способностью сетей передачи данных. Сбор технических данных приводит к значительному увеличению трафика в сети, что увеличивает вероятность отказа сети.

При обработке данных возникает проблема нехватки вычислительных ресурсов: ТВ-приставки, особенно старых образцов, имеют слабые технические возможности; локальные сервера, как правило, имеют достаточно низкую производительность. Решение задач по обработке данных приводит к существенному увеличению нагрузки на элементы СКЦТ, что, в свою очередь, влияет на функциональность пользовательских устройств, например, снижается скорость реакции ТВР, возникают ошибки при обработке команд.

Отсутствие возможности сбора необходимо объема данных и их обработки не позволяет существующим системам мониторинга своевременно выявлять ошибки, их локализовывать и устранять. Решение задач мониторинга и управления сетями возлагается на службы поддержки сетей.

Для возможности решения задач мониторинга и управления современными СКЦТ требуются системы, которые способны определять состав собираемых данных, обеспечивать их сбор и обработку, строить на основе результатов обработки модели, отражающие состояние сетей и отдельных их элементов. В таких системах состав собираемых данных должен быть обоснован, процессы построения моделей иметь низкую вычислительную сложность.

## **П.2. Исходные данные для построения моделей объектов по данным мониторинга в области телекоммуникаций**

В составе данных, содержащих информацию о состоянии кабельной сети, можно выделить следующие группы.

1. Сообщения об ошибках ТВР. Оперативные данные, представляющие собой значения отдельных параметров, оповещающие о возможности возникновения проблемной ситуации. Отдельные параметры могут обрабатываться на уровне программного кода. Как правило, обработка сводится к выводу на экран пользовательского сообщения об ошибке и/или отправке сообщения на сервер. Информация об ошибках передается на сервер сразу после их возникновения или с минимальной задержкой, определяемой логикой работы ТВР. В качестве примеров ошибок можно привести:

- «Ошибка EPG» – не получены данные EPG (Electronic Program Guide), содержащую программу передач, пользователь не может выбрать нужный канал;
- Ошибка «Черный экран» (устоявшийся термин «BlackScreen») – нет видеосигнала, пользователь видит только черный экран;
- «Ошибка DVR» – ошибка локальной записи, возникающая при попытке создать/сохранить DVR - запись (запись Digital Video Recorder);
- «Ошибка AVN» - пользователь не может просматривать AVN-канал (канал ActiveVideo Networks) ;
- «Ошибка SDV» - пользователь не может просматривать SDV-канал (канал Switched Digital Video);
- «Перезагрузка ТВР» - (устоявшийся термин «Reboot») произошла внезапная перезагрузка ресивера;
- «Фатальный сбой системы» - событие, после которого дальнейшее функционирование ресивера невозможно без перезагрузки;
- «Ошибки в процессе загрузки» - различные ошибки, возникающие в процессе запуска ТВР.

2. Параметры состояния аппаратных средств сервера и параметры работы сети (взаимодействие с конечными пользовательскими устройствами). Для их сбора разрабатываются отдельные сервисы. К предоставляемым параметрам состояния аппаратных средств относятся загрузка процессоров центрального или локальных серверов, размер занятой и свободной памяти, объем свободного места на конкретном жестком диске или разделе (например, разделе, где хранятся полученные лог файлы), объем трафика сети и другие. Работа сети характеризуется количеством запросов в секунду на некоторый



системный или пользовательский сервис, общим количеством запросов на некотором интервале времени, временем выполнения запросов, ошибками, возникающими в процессе выполнения запросов по типам сервисов, в том числе, с распределением по кластерам.

3. Лог файлы – буферные или исторические. Лог файлы представляют собой системный журнал работы технического средства. Буферные лог файлы выводятся через буфер в UDP или TCP порты. Исторические лог файлы – это файлы с наиболее важными лог записями, которые сохраняются на ТВР. Они могут быть получены по запросу. Возможно определить логические правила, по которым лог файлы будут выгружаться с устройств, например, “привязать” получение лог файлов к некоторым событиям. Поддерживаются несколько уровней логирования. В зависимости от уровня в лог файлы может выводиться информация об ошибках, информационные сообщения, отладочные сообщения и т.д. Такие файлы могут формироваться как на стороне ТВР, так и на стороне сервера. Файлы с логами могут храниться на центральном сервере или на выделенных серверах в сыром или в обработанном виде.

4. Краш-дампы, или пакеты логов специальных форматов, которые генерируются ТВР в момент фатального отказа, после которого происходит перезагрузка. Краш-дампы могут отсылаться на сервер автоматически, либо оператор может удаленно забирать дампы непосредственно с ТВР. Дампы хранятся и обрабатываются аналогично буферным логам, в виде файлов.

5. Телеметрические данные - это множество параметров, характеризующих состояние пользовательского устройства. Такие данные могут передаваться по запросу или выгружаться на сервер по расписанию. Можно привести следующие примеры телеметрических данных:

1) Одиночные системные сообщения (Heartbeats), которые могут посылаться с ТВ ресиверов на сервер с заданной частотой и использоваться для определения текущего состояния ТВР.

2) Периодически отправляемые конфигурируемые наборы системных параметров.

3) Данные о поведении пользователей - информация о просматриваемых каналах, длительности пребывания пользователя на канале, составе дополнительных запросов пользователя и т.д. Обычно эти данные накапливаются, агрегируются и используются в аналитических целях. Примеры регистрируемых событий: включение устройства, выключение устройства, переключение на другой канал, перезагрузка устройства, вывод сообщения об ошибке, запросы на выполнение пользовательских функций, запросы на получение данных о состоянии устройства и многие другие. Массивы собираемых данных имеют очень большой объем, число измеряемых параметров исчисляется сотнями.

4) SNMP-параметры, получаемые специальным сервисом путем разовых или периодических опросов ТВР. Данные передаются по SNMP-протоколу, разработанному для удаленного получения диагностических параметров и управления пользовательскими устройствами. Примеры SNMP-параметров: общий объем памяти ТВР и объем свободной памяти, объем флеш памяти, объем графической памяти и т.д. Общее число таких параметров, как правило, составляет несколько сотен. Список получаемых параметров может быть, при необходимости, расширен. Однако, это требует поддержки соответствующих параметров программными платформами устройств.

6. Данные об управлении абонентским оборудованием, передаваемые через глобальную сеть. Такие данные представляют собой нотификации, сформированные в соответствии со стандартом TR-069. Используемый для передачи нотификаций протокол SNMP позволяет выполнять следующие действия: начальная настройка устройства при его загрузке и внесение изменений в настройки уже работающего устройства. Удалённо могут выполняться и многие другие операции, в частности, обновление ПО; доступ к лог файлам и счётчикам производительности; диагностика состояния устройства; выполнение хранимых процедур. Отметим, что такие нотификации практически не используются. Это связано с тем, что поддержка протокола требует затрат значительных дополнительных ресурсов. Вместо нотификаций широко используется рассылка широковещательных сообщений через карусели на группы ТВР или специализированные частные интерфейсы, позволяющие выполнять отправку команд на отдельные устройства.

7. Сообщения о произведенных действиях «самолечения» ТВР. Такие процедуры, как правило, определяются логическими правилами, их выполнение может сопровождаться дополнительной информацией, в частности, формированием лог файлов. Примеры действий «самолечения»: освободить память, остановить / запустить сервис или приложение, произвести перезагрузку ресивера.

8. Диагностические страницы ТВР. Это перечень параметров о состоянии устройства, который выводится пользователю на экран телевизора. Они организованы в виде последовательности страниц. Отображаться может информация обо всех параметрах либо какие-то определенные параметры (по их номеру). Те же самые страницы могут быть переданы на сервер. Они отсылаются в текстовом виде, по запросу. Можно задать периодическую автоматическую доставку параметров.

В целом, состав исходных данных является достаточным для решения задач мониторинга и управления СКЦТ.

### **II.3. Результаты построения моделей объектов по данным мониторинга в области телекоммуникаций**

При синтезе моделей объектов сетей кабельного телевидения для мониторинга и управления такими сетями интерес представляют параметры состояния элементов этих сетей, их поведение и связи между ними.

При синтезе моделей объектов СКЦТ учитывались следующие особенности:

1. Отсутствие систематизированной классификации ошибочных ситуаций. Как правило, описываются только крупные группы ошибочных ситуаций. Такое описание чаще всего не позволяет локализовать ошибку, оперативно определить причину ее возникновения.

2. Наличие большого числа параметров и множественных связей между ними. Количество параметров элементов сетей может составлять сотни. Связи могут существовать как между парами элементов, так и между многими элементами. Большинство из этих связей не известны.

3. Информация поступает от многих источников. Возникает необходимость согласования информационных потоков.

4. Информация плохо структурирована и слабо формализована. Основная причина этого заключается в отсутствии стандартов на форматы информационных сообщений. Такая ситуация наблюдается и для текстовых сообщений и для потоков результатов измерений.

С учетом перечисленных особенностей при построении моделей СКЦТ решались задачи выявления зависимостей между параметрами различных элементов сети, определения сценариев поведения пользователей и их влияния на значения параметров.

При выявлении взаимосвязей между параметрами состояния ТВ-ресиверов учитывалась их многоуровневая структура. Связи определялись на уровне программных платформ, уровне компонентов и модулей, уровне значений отдельных параметров.

При восстановлении зависимостей между параметрами рассматривалось несколько типов таких зависимостей, в частности, зависимости между параметрами состояния и их поведением компонентов ТВР и серверных компонентов. Выявлялись зависимости, которые описывают штатное поведение СКЦТ, а также зависимости, наблюдаемые при возникновении ошибочных ситуаций в сети.

Осуществлялся поиск зависимостей между группами элементов, а также зависимостей, связывающих отдельные элементы.

*Результаты восстановления зависимостей между параметрами ТВР и параметрами серверных компонентов.* В таблице II.1 приведены отдельные зависимости, выявленные между параметрами ТВР и параметрами серверных компонентов.

Таблица П.1 – Зависимости между параметрами ТВР и параметрами серверных компонентов

STB Problem	STB Indicator	STB General Cause	Detailed Cause	Backend or AMS Problem	AMS Indicator
Канал не доступен / не авторизован (Novideo / BlackScreen)	Popup "Channel unavailable"	LineUp is absent or corrupted	channel's source_id doesn't exist	AMS Lineup service problem, AMS Publisher problem	LIN-001, LIN-002, LIN-004, LIN-009, LIN-023, LIN-030, LIN-031, LIN-032 etc
			Invalid channel attributes in Lineup	AMS Lineup service problem, AMS Publisher problem	LIN-*, PBL-*
		No first frame	PAT is unavailable (PID #0)	MSO Headend problem	
			PAT is available, but "mpeg program number" is not specified in it	MSO Headend problem	
			PAT is available, but PID number (PMT) is not available in the stream	MSO Headend problem	
		Video format is not supported	no MPEG4 codec	VOD HE problems; billing problems;	
	Popup "Instant upgrade"	channel status = "not authorized"		lineup or packages middle micro-service problems; billing problems (CSG/DAC/etc.)	ENT-*
Не работает VOD		VOD asset was purchased but can't be displayed			
Не работает PPV		PPV asset was purchased but can't be displayed		ppv metadata missing; ppv notification service problems; ppv authorization missing on the controller	PPV-*
Отсутствует программа передач (No EPG data error)	EPG data download failed		File is not exist.	Access to Broadcaster or DNCS or HTTP was broken	PBL-*
			Can't load file.		
			Invalid file version was passed.		
			Insufficient of the memory.		
			Timeout happens while loading file.		

STB Problem	STB Indicator	STB General Cause	Detailed Cause	Backend or AMS Problem	AMS Indicator
			Can not parse loaded data.		EPG-*
	Popup Guide data Unavailable				
	Unable to schedule	program data is unavailable			
DVR Errors	Unable to Schedule		There is no access to AMS server		DVR-*
	Unable to Save		There is no access to AMS server		DVR-*
	Unable to delete		There is no access to AMS server		DVR-*
Ресиверы не могут зарегистрироваться на сервере (Boxescann'tregisteronAMSregion)	migration_flag 0; registration_flag 0;	information pertaining to the boxes is missing from the relevant AMS DB tables	ams is unreachable; network problems; wrong AMS IP in profile; AMS VIP problems;	Problem with MSO backend	
Незавершается миграция Settings (Settings migration was not completed)	migration_flag 0; registration_flag 0;	information pertaining to the boxes is missing from the relevant AMS DB tables	AMS <-> Settings Cassandra	Problem with MSO backend	SET-*
Незавершается миграция DVR (DVR migration can't finish successfully)	migration_flag 0; registration_flag 0;	information pertaining to the boxes is missing from the relevant AMS DB tables	AMS <-> DVR Cassandra communication is broken;	Problem with MSO backend	DVR-*

Таблица П.2 – Возможные ошибки серверных компонентов

Level	Code	Log Message Template	Description
Error	LIN-001	Cannot initialize CharterLineupService <error details>	Error initializing lineup service or incorrect configuration file.
Error	LIN-002	LineupModuleException <error detail>	Error in the Lineup service.
Error	LIN-004	JSON lineup input data file parsing error <error details>	Error in the input lineup JSON data file.
Error	LIN-005	Cannot download logo for channel <error details> . Communication error has occurred	Channel logo cannot be downloaded due to communication error with logo service.
Error	LIN-009	Duplicate channel number found in <channel metadata>	Incoming lineup contains duplicate channel number
Error	LIN-020	URL for channel logo updating is missing!	No URL to Tribune logo feed is defined
Warning	LIN-003	Cannot download logo for channel <error details>. The channel logo image is missing	Either the channel logo image is missing or a communication error has occurred.

Level	Code	Log Message Template	Description
Warning	LIN-006	Error obtaining logo crc: <error details>	Problems obtaining logo image
Warning	LIN-007	Error uploading logo image to a database: <error details>	Problems uploading logo image to Lineup Service DB
Warning	LIN-008	Error getting logo image from a database: <error details>	Problems obtaining logo image from Lineup Service DB
Warning	LIN-019	Cannot convert image: <error details>	Problems converting logo image
Warning	LIN-022	Downloaded image file size is 0 for {TMS service ID}	Logo image file size is 0 for the channel ..
Warning	LIN-023	Lineup Module Exception. <error details>	Service Exception
Warning	LIN-024	Channel {channel metadata} EAS parameters are incomplete	EAS channel parameters are not sufficient
Warning	LIN-026	Channel skipped: inconsistent or missing lineup data for channel { }	Channel was skipped due to inconsistent or missing lineup data
Warning	LIN-027	CALLSIGN is missing for channel { }	CALLSIGN field information is missing or empty for channel
Warning	LIN-028	NUMBER is missing for channel { }	NUMBER field information is missing or empty for channel
Warning	LIN-029	TMSERVICEID is invalid for channel { }	TMSERVICEID field information includes not only numbers or number is out of range
Warning	LIN-030	TYPE is invalid for channel { }	TYPE field information is invalid, missing or empty for channel
Warning	LIN-031	PARAMETERS is invalid for channel { }	PARAMETERS field information is invalid, missing or empty for channel
Warning	LIN-032	CASID is invalid for channel { }	CASID field information is missing or empty for channel
Warning	LIN-033	ISHD is invalid for channel { }	ISHD field information is invalid or missing for channel
Warning	LIN-034	ISPREMIUM is invalid for channel { }	ISPREMIUM field information is invalid, missing or empty for channel
Warning	LIN-035	SDHPAIR is invalid for channel { }	SDHPAIR field information is present and invalid for channel
Warning	LIN-036	Invalid HUBID for <lineup data>	Hub id is invalid
Warning	LIN-037	There is no LINEUP data for box type { }	Lineup data for the specified box type is missing from the database
Warning	LIN-038	Duplicate LINEUP hub id is found: { }	Hub id is duplicate
Warning	LIN-039	PARAMETERS field is invalid for channel { }	PARAMETERS field of the lineup feed is incorrect
Info	LIN-014	Lineup service is waiting...	Service is in sleep mode
Info	LIN-015	Lineup service is scheduled to run at {date/time}	Service will be launched on ..
Info	LIN-018	Lineup service is activated.	Service is activated.
Info	LIN-021	{number} logos updated, {number} logos missing.	Statistics on logo images updat

Таблица П.3 – Примеры отдельных зависимостей, выявленных по данным сегмента Рено

Description	STBLOG	AMSLOG	Resolution
Can'tschedule	Cloud DVR interaction:	2016-07-21 05:52:11.338 DEBUG http-bio-8080-exec-23-	Different

Description	STBLOG	AMSLOG	Resolution
recording from CloudGuide	<p>Line 8020: *403.60 4A213A90 {N} AmsUtils: msg_num: 1 [real: 0]  Line 8021: *403.60 4A213A90 {N} AmsUtils: mac_address: 0000028529ee  Line 8022: *403.60 4A213A90 {N} AmsUtils: message: 0, length: 47, type: 0x7E  Line 8023: *403.60 4A213A90 {N} AmsUtils: standart message:  Line 8024: *403.60 4A213A90 {N} AmsUtils: flags 0x2048  Line 8025: *403.60 4A213A90 {N} AmsUtils: <b>sender is dvr/3</b></p> <p>Cloud DVR incoming request for scheduling:  Line 9171: *453.82 4A216790 {N} AmsUtils: msg_num: 1 [real: 1]  Line 9172: *453.82 4A216790 {N} AmsUtils: mac_address: 0000028529ee  Line 9173: *453.82 4A216790 {N} AmsUtils: message: 0, length: 36, type: 0x7E  Line 9174: *453.82 4A216790 {N} AmsUtils: standart message:  Line 9175: *453.82 4A216790 {N} AmsUtils: flags 0x2009  Line 9176: *453.82 4A216790 {N} AmsUtils: sender is dvr  Line 9177: *453.82 4A216790 {N} AmsUtils: addressee is <b>dvr/1</b></p>	<p>0000028529EE/DvrSenderToStbExt - [sendMessage] message:  DOBTransportMessage{ deliveryInfo=DOBDeliveryInfo{ stbId='0000028529EE', ip=0.0.0.0,<b>stbServiceId='dvr/1'</b>, serviceId='dvr', messageType=0, transportId=778, transactionId=0, errorCode=ERR_OK, timeout=120000, responseExpected=true, isFromBox=false, bodyType=BYTES, customProperties=[], dataMessage=null, data=[04, 01, D5, 57, 57, 90, B2, 38, 88, 0E, 05, 01, 00, 00, 00, 00, 1A]}</p>	version support on STB and AMS
Timeout detected by BoxResponseTracker	<p>03/12@05:10:01 - 10.0.71.181 : &gt;&gt;&gt; z_powerup[101]: [I] RUDP: <b>Received packet, from: ip = 172.30.140.80</b>, port = 23693  03/12@05:10:01 - 10.0.71.181 : &gt;&gt;&gt; z_powerup[101]: [I] RUDP: 0000 : 03 01 01 90 01 F3 47 2D  03/12@05:10:01 - 10.0.71.181 : &gt;&gt;&gt; z_powerup[101]: [I] RUDP: <b>sent start packet: ip = 172.24.133.99</b>, port = 7750, id = 9203, timestamp = 196757</p>	<p>2015-03-12 05:14:55.994 ERROR pool-4-thread-39/DvrModuleMainImpl - DVR-010 Message from STB processing error: Timeout detected by BoxResponseTracker</p>	Incorrect RUDP server IP was set on the STB
Unable to schedule		2016-09-07 18:43:22.851 WARN New I/O worker #146/RUDP1Handler -	STB losts

Description	STBLOG	AMSLOG	Resolution
<p>program from AVN due this error in RUDP</p>	<pre>RUDP: 0000 : 0101-5EAD 0132        ..^..2 RUDP: process_start_pack: segment size = 350, file id = <b>173</b>, file size = 50 RUDP: sent ack start packet: ip = 172.24.133.99, port = 7750, id = 173, timestamp = 1199849 RUDP: data pack dump: RUDP: 0000 : 0301-015E 05AD-0132        ...^..2 RUDP: Received packet, from: ip = 172.30.140.80, port = 42593 RUDP: 0000 : 0491-DC05 AD01            ..... <b>RUDP: process_error_pack: server doesn't recognize file_id.</b> Cancel transfer</pre>	<pre>RUDP: cannot retrieve session for ACK-START packet: AckStartPacket{maxSegmentQtyPerSession=5, segmentSize=350, fileId=22370, dataSize=50}, key: SessionKey{remoteAddress=/10.10.65.173, fileId=22370, hashCode=38245494}, address: /10.10.65.173:7760 2016-09-07 18:43:22.851 WARN New I/O worker #146/RUDP1Handler - RUDP: cannot retrieve session for ACK-START packet: AckStartPacket{maxSegmentQtyPerSession=5, segmentSize=350, <b>fileId=22370</b>, dataSize=50}, key: SessionKey{remoteAddress=/10.10.65.173, fileId=22370, hashCode=38245494}, address: /10.10.65.173:7760 2016-09-07 18:43:22.851 DEBUG New I/O worker #146/RUDPSendManager - RUDP: Add message to sender queue: WrongSessionIdErrorResponsePacket{remoteAddress=/10.10.65.173:7760, fileId=22370}</pre>	<p>session ID</p>
<p>DC messages from Cisco boxes are not posted to .csv files</p>	<pre>... 04/28@05:20:33 - 10.0.71.39 : &gt;&gt;&gt; z_powerup[101]: [N] APP: service_lib_type::is_service_authorized: -&gt; 1 04/28@05:20:33 - 10.0.71.39 : &gt;&gt;&gt; z_powerup[101]: [I] ScriptLogger: [start] 1135189 [D][NAV][AUTH]"POLL" checkAuthorization ch:2 servId:2 -&gt; 1 (0x0000) 04/28@05:20:37 - 10.0.71.39 : &gt;&gt;&gt; Supervisor[100]: [I] BrickCAS: IsAuthorizationCheckingEnabled () 04/28@05:20:38 - 10.0.71.39 : &gt;&gt;&gt; z_powerup[101]: [N] APP: service_lib_type::is_service_authorized: -&gt; 1 04/28@05:20:38 - 10.0.71.39 : &gt;&gt;&gt; z_powerup[101]: [I] ScriptLogger: [start] 1140244 [D][NAV][AUTH]"POLL" checkAuthorization ch:2 servId:2 -&gt; 1 (0x0000)</pre>	<pre>... 2015-04-28 05:20:30.939 WARN New I/O worker #146/RUDP1Handler - RUDP: unknown packet type, remote address: /172.24.133.102:7495, bytes: [100, 101, 102, 97, 117, 108, 116, 32, 115, 101, 110, 100, 32, 115, 116, 114, 105, 110, 103], sending upstream w/o rudp decoding 2015-04-28 05:20:30.948 DEBUG New I/O worker #146/ZodiacNettyTransport - Unsupported Zodiac message type: 102 2015-04-28 05:20:30.948 DEBUG New I/O worker #146/ZodiacNettyTransport - Error while reading the message; ignored</pre>	



Зависимости исследовались с точки зрения их влияния на пользовательские функции конечных устройств. Некоторые ошибки, регистрируемые на сервере, приведены в таблице П.2, детальное описание ошибок приводится в технической документации по серверным решениям. В таблице П. 3 приведены примеры отдельных зависимостей, которые были определены по данным, полученным по району Рено (Reno, USA).

*Результаты восстановления зависимостей между группами ошибок.* При восстановлении зависимостей между группами ошибок использовалась БД о состоянии сегмента сети, развернутого в районе Рено, объем которой составлял ~1,25 млн. записей. На основе этих данных была сформирована выборка объемом 47050 записей лог файлов сервера. В выборке содержались ошибочные сообщения различных типов: ошибки, зарегистрированные на сервере (ams\_log), ошибки "черный экран" (bs\_err\_type), ошибка DVR (dvr\_type), ошибка AVN (no\_avn\_type), ошибка «Отсутствует программа передач» (no\_epg\_type), ошибка SDV (sdv\_err\_type). Число ресиверов, по которым были зафиксированы ошибки, составило 98 штук.

Распределение количества ошибок по типам представлено на рисунке П.2, численные значения приведены в таблице П.4.

Таблица П.4 – Распределение количества ошибок по типам на основе данных по сегменту сети Рено

Тип ошибки	ams_log	bs_err_type	dvr_type	no_avn_type	no_epg_type	sdv_err_type
Число ошибок	486	12720	61	5035	3515	25233

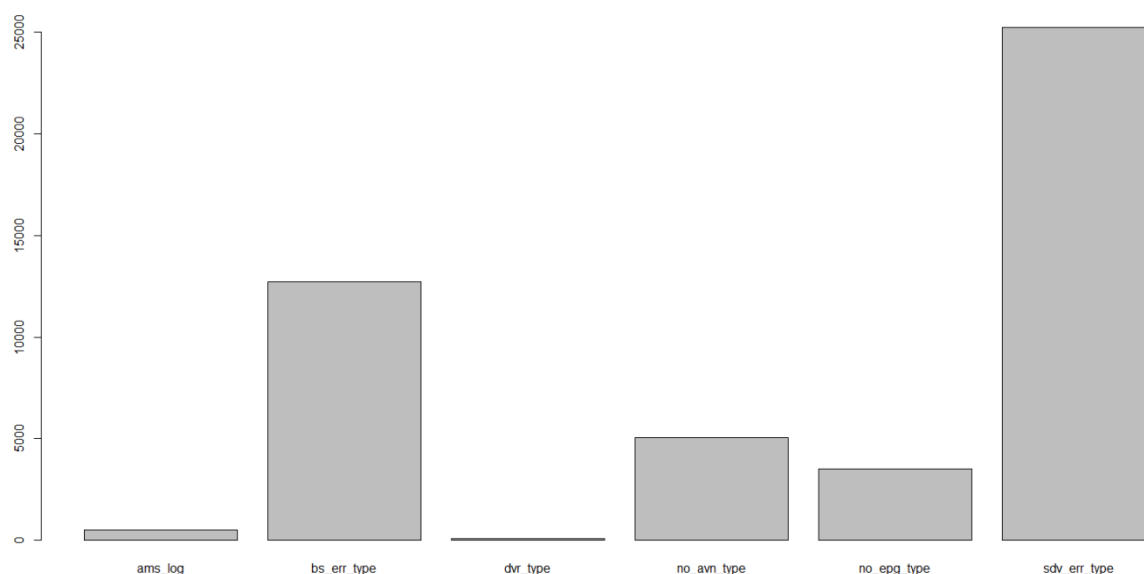


Рисунок П.2 – Распределение количества ошибок по типам на основе данных по сегменту сети Рено

Для выявления зависимостей было построено распределение типов ошибок в зависимости от времени их регистрации на устройствах. Информация была получена из лог файлов сервера и из БД ошибок ТВР. Построенная зависимость показана на рисунке П.3. На рисунке П.3 по оси ординат слева перечислены виды ошибок (событий), справа – распределение ошибок во временном интервале (вероятность от 0 до 1 появления ошибки в некоторый момент времени). Ось абсцисс – это временная шкала в условных единицах. Например, видно, что до момента 200000 условных временных единиц ошибки, зафиксированные в серверных лог файлах (всплеск темно-серого цвета), практически отсутствовали, с наибольшей вероятностью такие ошибки встречались на интервалах [200000 – 250000] и [390000 – 410000].

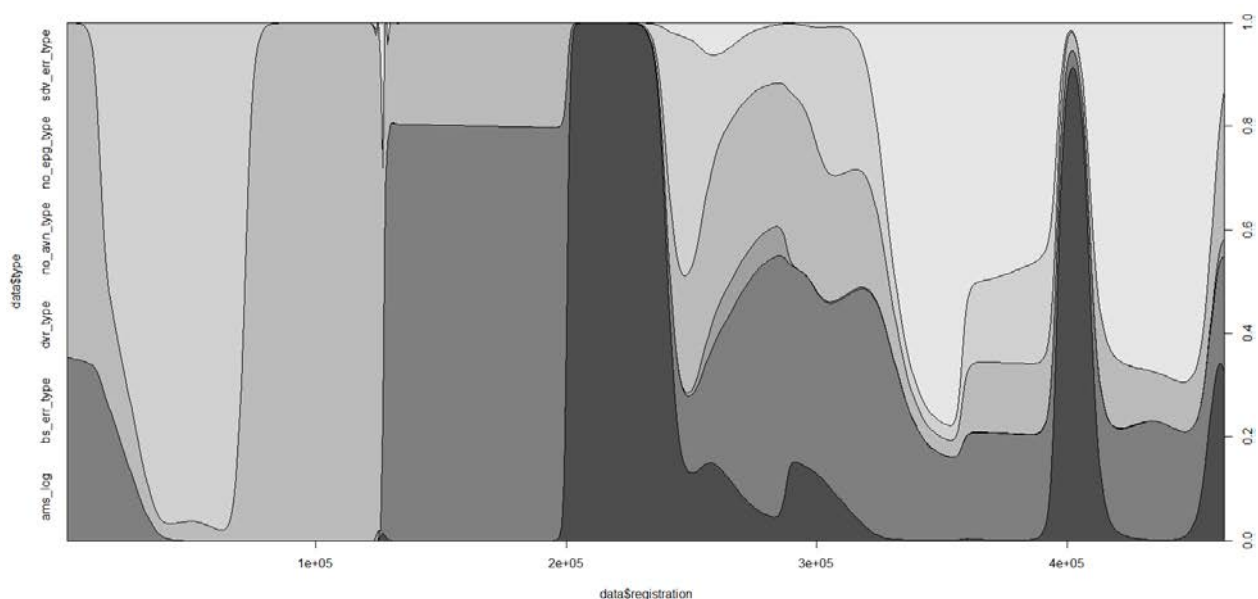


Рисунок П.3 – Зависимость времени регистрации ошибок от их типа

Для оценки особенностей появления ошибок каждого типа были рассчитаны характеристики, приведенные на рисунке П.4. Диаграмма на рисунке П.4 показывает медиану и квартили (Q25, Q50, Q75, Q95), межквартильный размах (IQR), минимальное и максимальное значение количества ошибок и выбросы по каждому типу ошибок.

Построено также трехмерное распределение, характеризующее зарегистрированные события (рисунок П.5). Распределение отражает зависимости между следующими параметрам: типы ошибок, mac-адреса устройств и время регистрации событий. При представлении типов данных используются следующие обозначения: 1 – записи лог файлов сервера (точки черного цвета), 2 – ошибка типа bs\_err\_type (точки красного цвета), 3 – ошибка типа dvr\_type (точки зеленого цвета), 4 – ошибка типа no\_avn\_type (точки синего цвета), 5 – ошибка типа no\_erg\_type (точки голубого цвета), 6 – ошибка типа sdv\_err\_type (точки розового цвета). Из рисунке П.5 видно, что последовательность событий (возникновения ошибок) имеет определенную структуру.

Достаточно регулярное поведение последовательностей наблюдается для ошибок типов 2 (bs\_err\_type) и 4 (no\_avn\_type). Также видно, что события, соответствующие возникновению некоторых ошибок, имеют связи. Так, ошибки вида 2, 4 и 6 (sdv\_err\_type) возникают одновременно на большинстве устройств (mac-адресов).

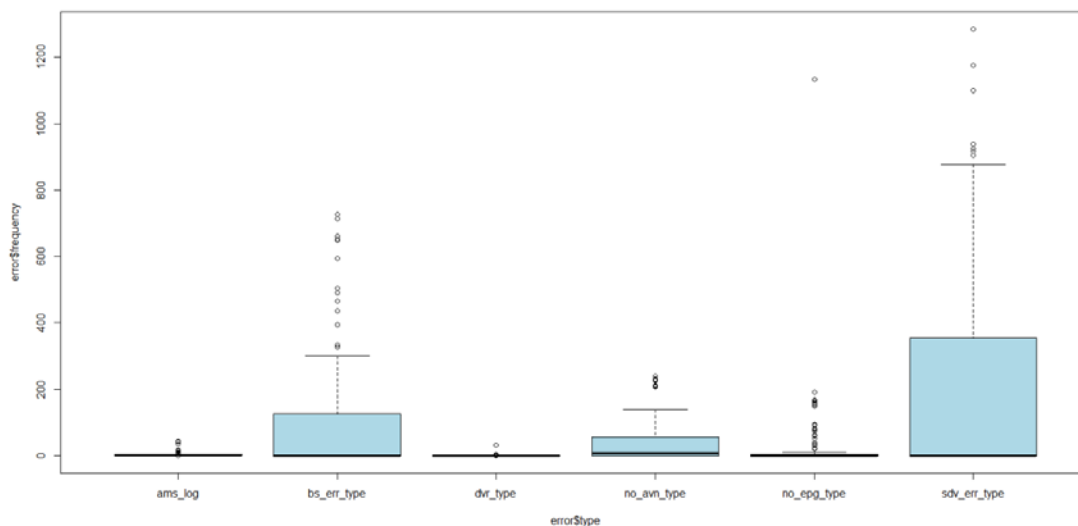


Рисунок П.4 – Характеристики ошибок различных типов

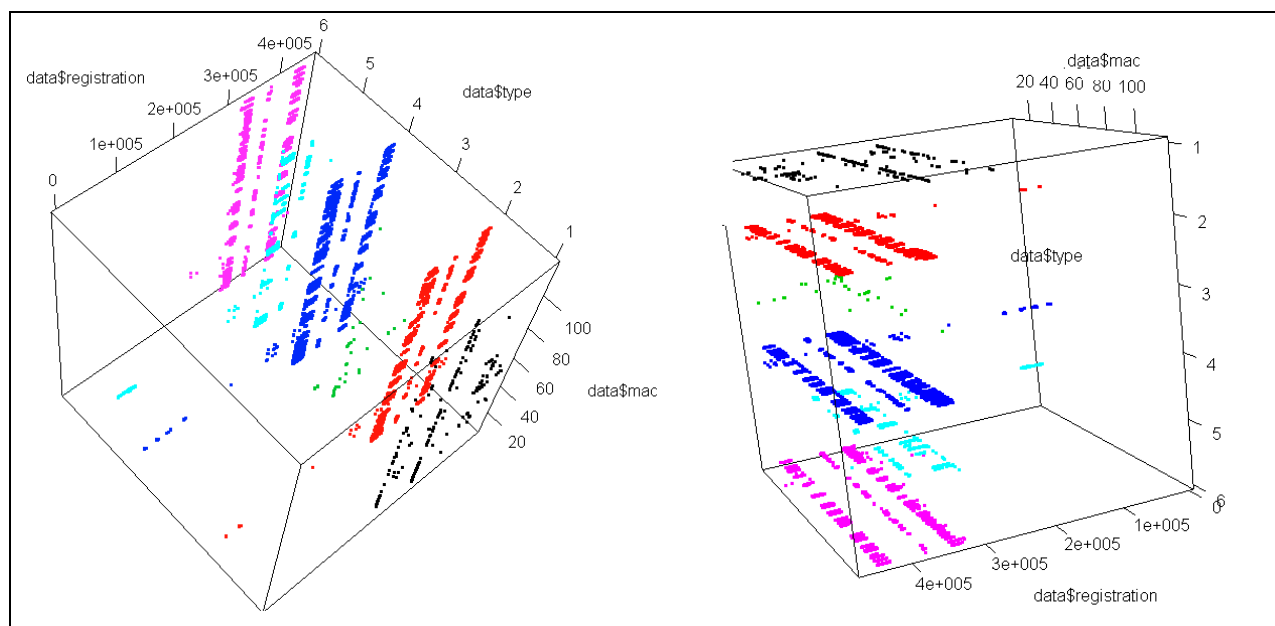


Рисунок П.5 – Распределение различных типов зарегистрированных ошибок по устройствам и времени их возникновения

*Результаты восстановления зависимостей для ошибок одного типа.* Рассмотрим восстановление зависимостей для ошибок одного типа на примере ошибок DVR. Исходными данными для синтеза модели являлись лог файлы сервера, включавшие 322 файла общим объемом ~6,5Гб за сутки. В данных лог файлов содержалась информация о порядка 25 млн. событий.

Состав исследуемых ошибок приведен в таблице П.5.

Таблица П.5 – Записи лог файлов сервера, связанные с ошибкой DVR

Level	Code	LogMessageTemplate	Description
Error	DVR-009	STB<STBMAC>synchronizationerror: <errordetails>	Datasynchronization error has occurred
Error	DVR-010	Message from STB processing error: <error details>	Error processing a message received from an STB
Error	DVR-011	Cannot set priority vector: stb <STB MAC> not found in the database	An STB is not found in the database
Error	DVR-013	Shared Library error: <error details>	DVR Shared Library error
Warning	DVR-016	CHE message <message type> processing error: <error stacktrace>	Error processing a message received from headend

Зависимости восстанавливались на основе анализа одномерных и многомерных распределений:

- распределение количества событий по типам событий (типам ошибок DVR);
- распределение количества событий по времени их возникновения с учетом типов ошибок DVR;
- распределение событий по пользователям с учетом типов ошибок DVR;
- распределение событий по пользователям с учетом уровня сообщений, выводимых в лог файлы (Warn, Error) (рисунок П.6);
- распределение событий по пользователям с учетом типов ошибок DVR (рисунок П.7);
- распределение сообщений лог файлов по их уровням;
- распределение сообщений лог файлов по типам ошибок;
- распределение событий DVR по компонентам (таблица П.6) и процессам (таблица П.7) и др.

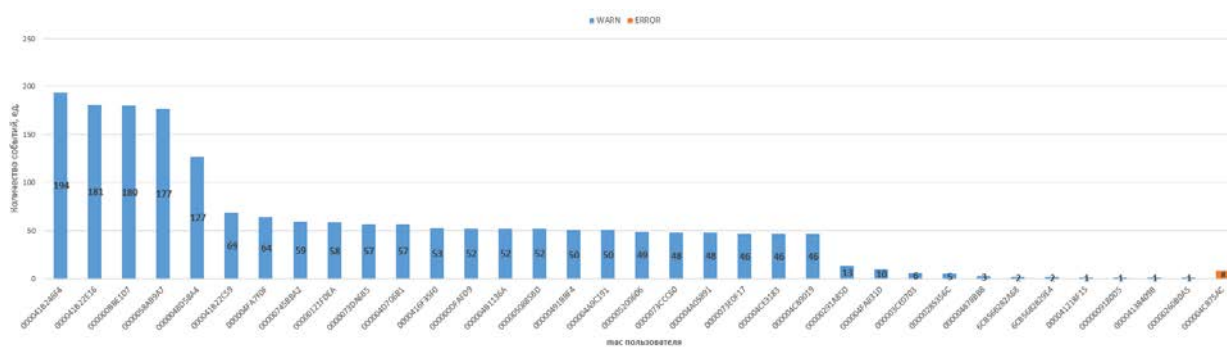


Рисунок П.6 – Распределение количества событий по пользователям с учетом уровня сообщений об ошибках (warn, error)

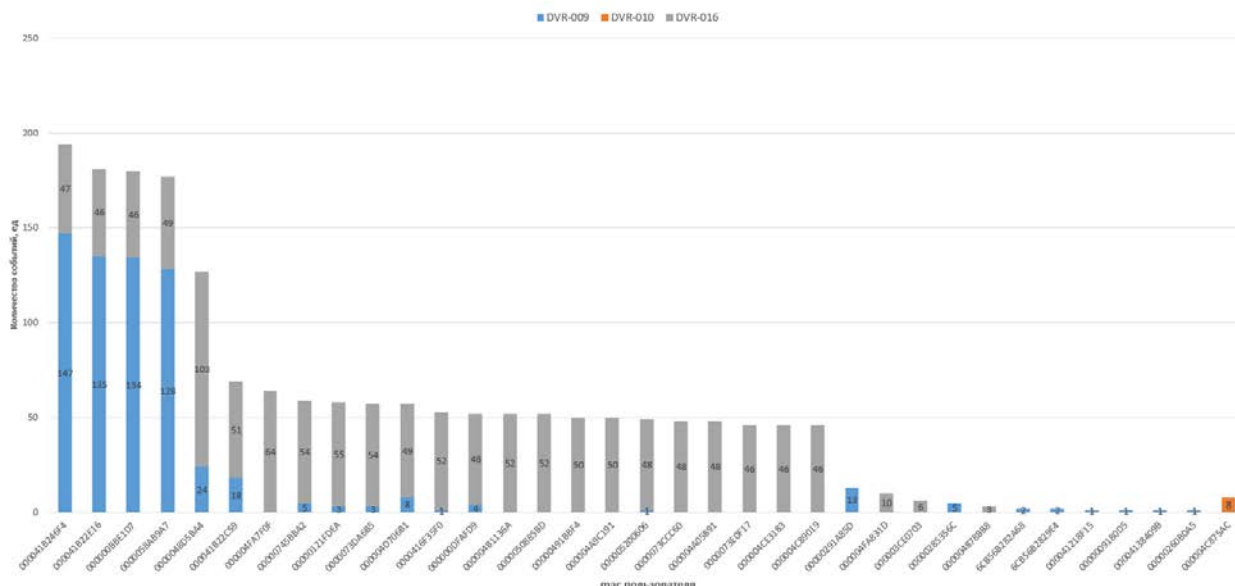


Рисунок П.7 – Распределение количества событий по пользователям с учетом типов ошибок

Таблица П.6 – Распределение событий DVR по компонентам

Компонент/модуль	DVR-009	DVR-010	DVR-016
DvrDeltaProcessorImpl	637	0	0
DvrHeadendReqProcessorImpl	0	0	1223
DvrModuleMainImpl	0	8	0

Таблица П.7 – Распределение событий DVR по процессам

Тред/процесс	DVR-009	DVR-010	DVR-016
http-bio-8080-	0	0	1223
pool-18-thread	0	8	0
pool-33-thread	637	0	0

Примеры выявленных зависимостей приведены на рисунках П.8 – П.9.

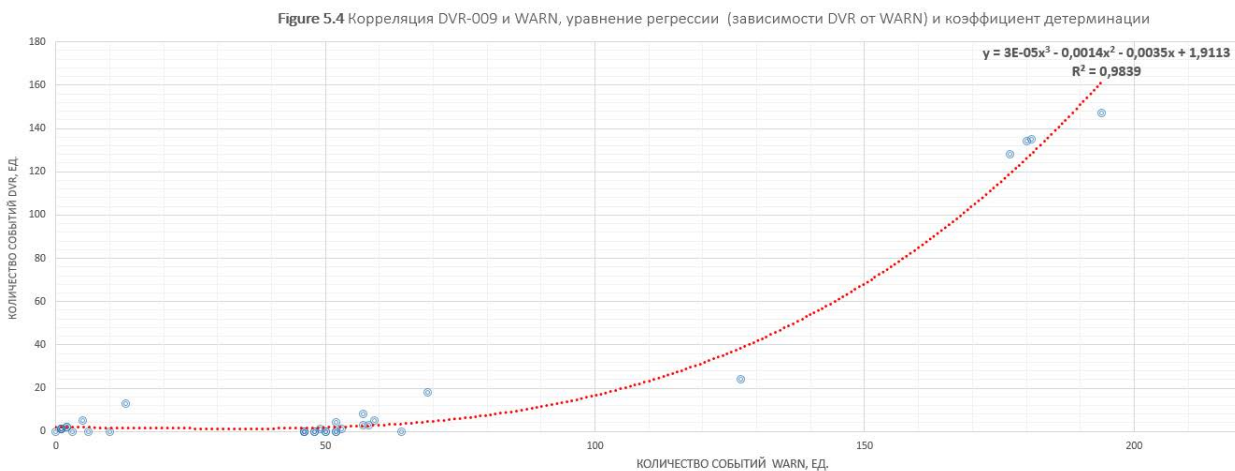


Рисунок П.8 – Регрессионная зависимость ошибок DVR-09 от уровня связанных сообщений лог файлов

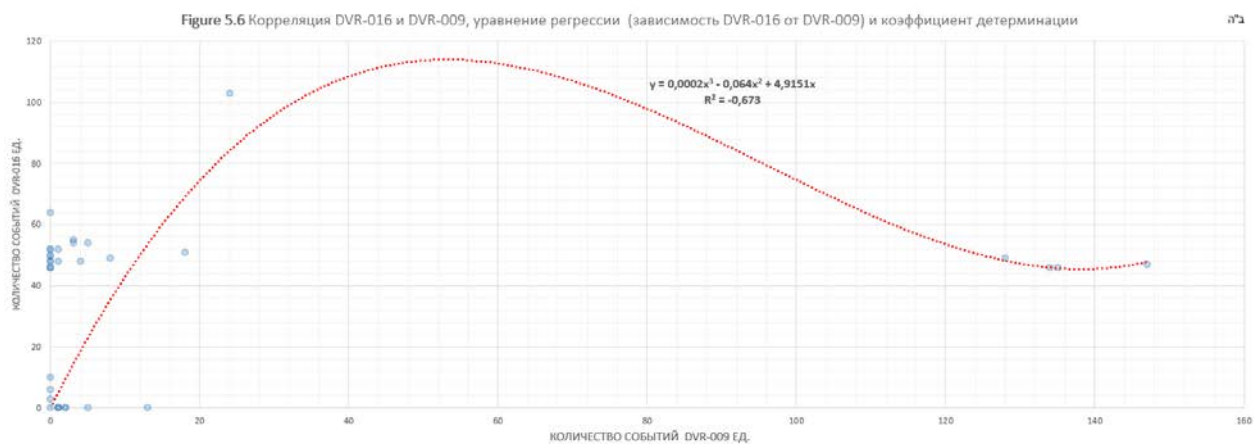
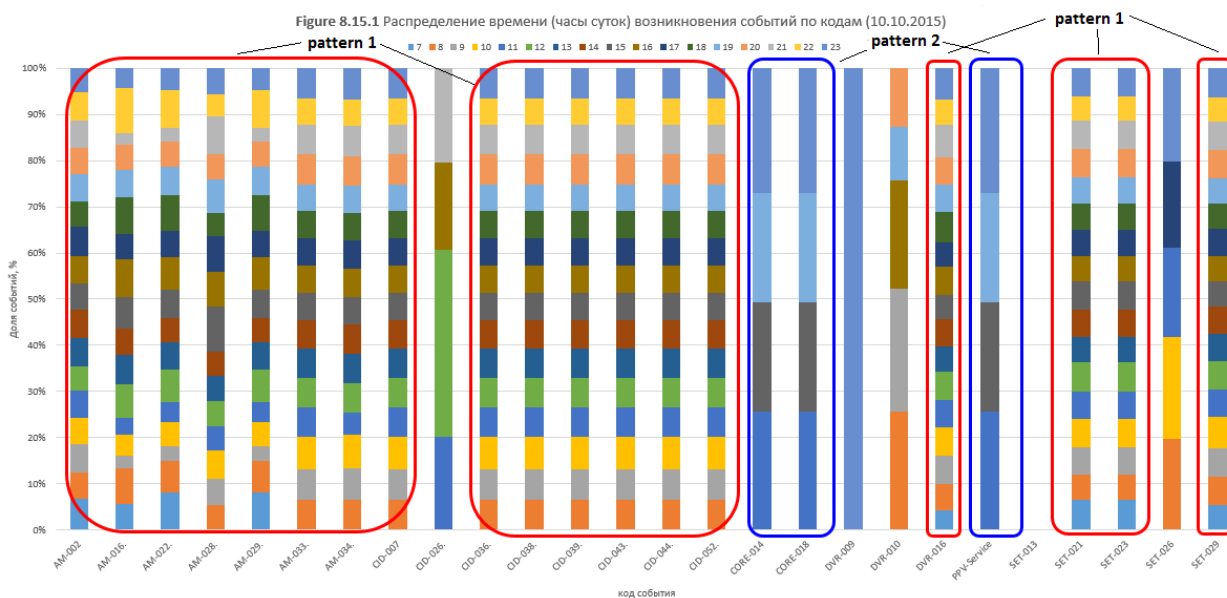


Рисунок П.9 – Регрессионная зависимость ошибок DVR-09 и DVR-016

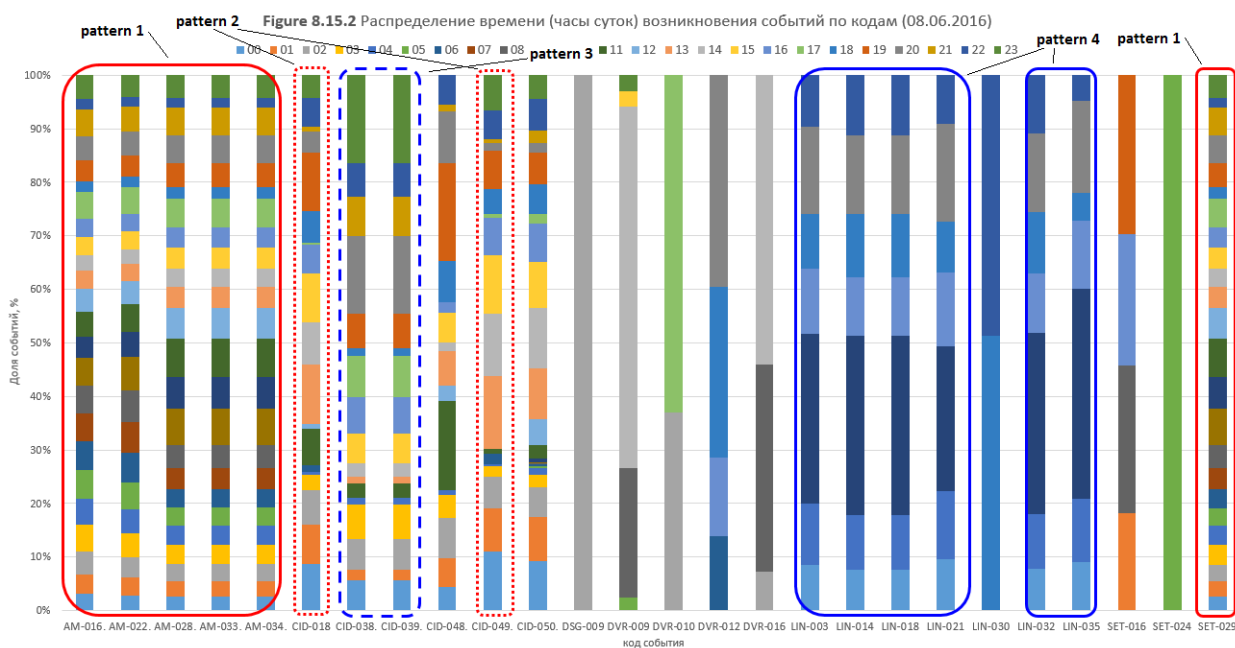
На графике рисунка П.8 представлена регрессионная зависимость ошибок DVR-09 от уровня сообщений лог файлов. Вычисленные коэффициенты детерминации (их значения приближаются к единице) показывают, что построенные регрессионные зависимости хорошо соответствуют моделируемым данным. Из рисунка П.9, видно, что коэффициент детерминации между числом возникающих ошибок DVR-009 и DVR-016 отрицательный. Это говорит о крайне слабой зависимости между ошибками этих типов.

Зависимости, выявленные между параметрами ТВР и параметрами серверных компонентов, а также характеризующие группы ошибок и ошибки отдельных типов наблюдаются при различных условиях, что позволяет их отнести к достаточно устойчивыми (постоянными) зависимостям.

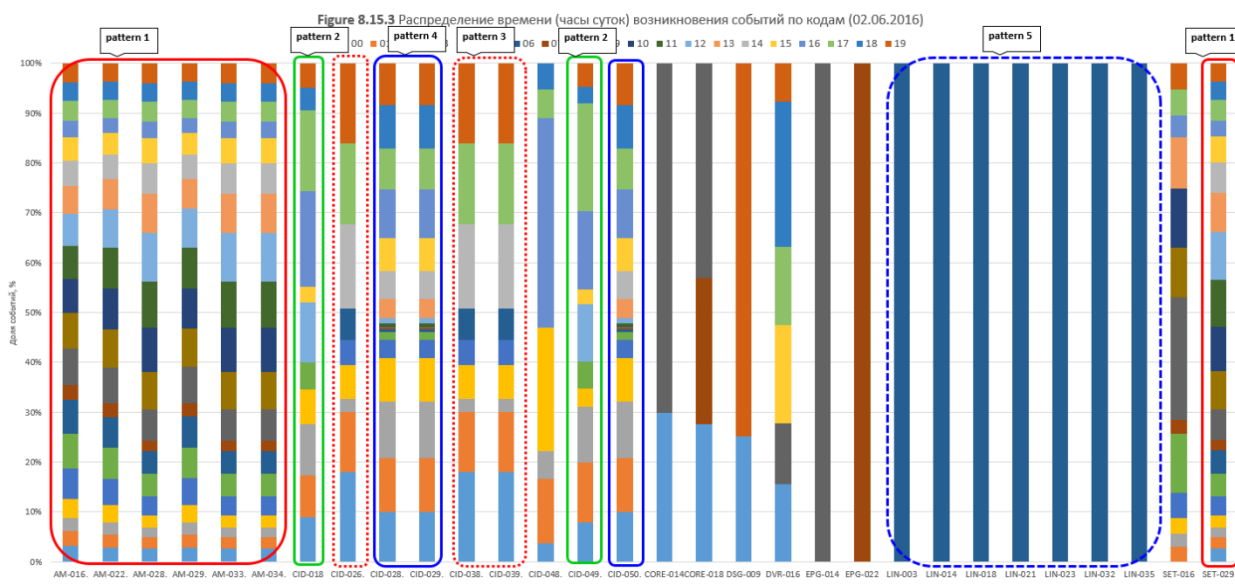
*Результаты восстановления зависимостей для ошибок AMS.* Рассматривались распределения различных типов ошибок AMS, зарегистрированных в сегменте сети Рено при различных условиях. На рисунке П.10 показаны примеры характерных распределений ошибок различных типов, наблюдаемых на суточных интервалах.



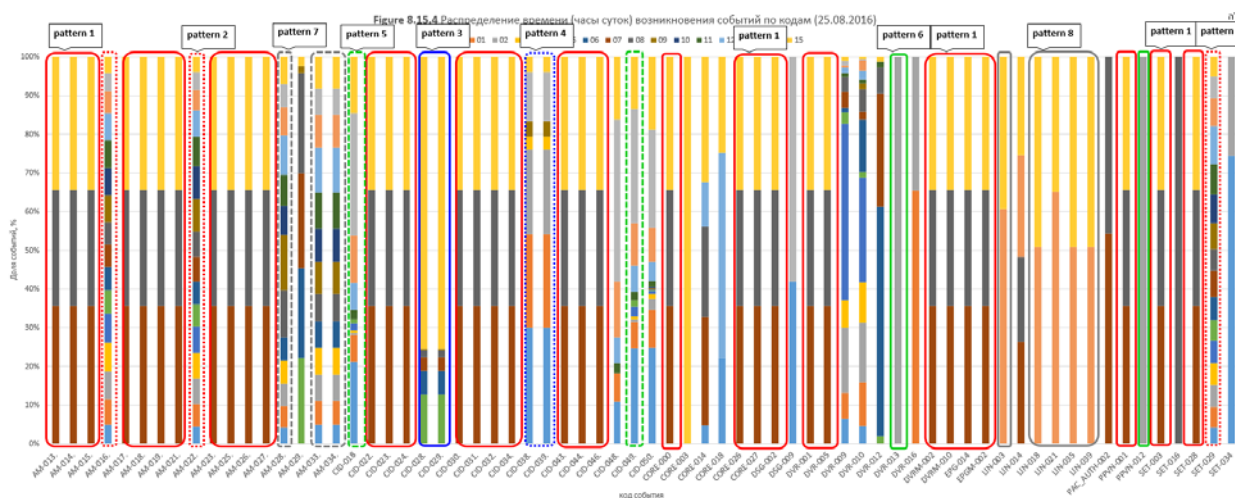
а) Распределение ошибок AMS по данным за 25.08.2016



б) Распределение ошибок AMS по данным за 08.06.2016



в) Распределение ошибок AMS по данным за 02.06.2016



г) Распределение ошибок AMS по данным за 25.08.2016

Рисунок П.10 – Характерное распределение ошибок AMS на суточных интервалах

Как правило, такие зависимости чувствительны к параметрам СКЦТ. В связи с этим они обычно определяются для каждого региона отдельно. С учетом этого, выявленные зависимости ошибок относятся к условно-постоянным зависимостям.

*Результаты восстановления зависимостей в поведении компонентов ТВР.* Зависимости в поведении компонентов ТВР описываются в виде шаблонов. Шаблоны представляют собой последовательность событий, регистрируемых в лог файлах пользовательских устройств. Они могут описывать как типовое поведение компонентов, наблюдаемое при выполнении различных пользовательских сценариев, так и ошибочное поведение. Шаблоны позволяют детально описывать зависимость в поведении компонентов ТВР. Ниже показаны построенные шаблоны поведения ТВР на примере ошибки "черный экран" (ошибка отсутствия изображения) и на примере ошибки DVR.

На рисунке П.11 показаны записи в лог файлах ТВР при наличии (верхняя диаграмма) и отсутствии ошибки "черный экран" (нижняя диаграмма). По оси x указано время, а по оси y – зарегистрированные сообщения. На рис. П.11 пики соответствуют этапу настройки устройства на канал и начала формирования изображения.



Рисунок П.11 – Записи в лог файлах ТВР, при наличии (верхняя диаграмма) и отсутствии ошибки "черный экран" (нижняя диаграмма)

В лог файлах могут содержаться сообщения, не связанные с исследуемой ошибкой. После удаления сообщений диаграмма приобретает вид, показанный на рисунке П.12.

С использованием диаграммы были построены шаблоны, соответствующие поведению компонентов ТВР при наличии и отсутствии ошибки "черный экран". Текстовое описание шаблона при успешном подключении на канал приведено на рисунке П.13. Графическое представление шаблона показано на рисунке П.14.



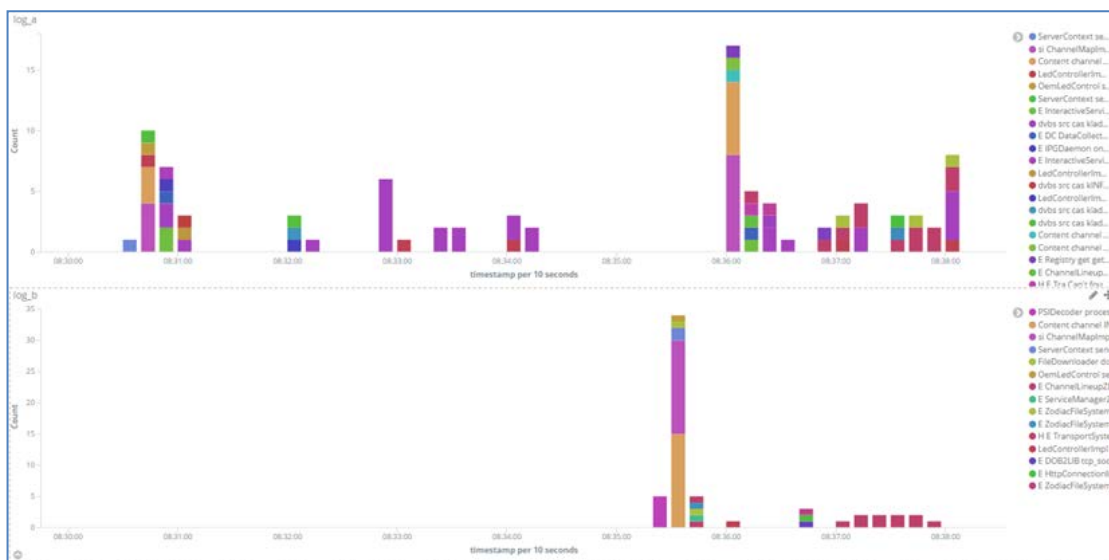


Рисунок П.12 – Записи в лог файлах ТВР, связанные с формированием изображения

The pattern of events in the log in case of successful tuning:

[INF] 01.12.2016 08:36:18.926 [DVBS\_VIDEO] [615:898 MediaPresentationImpl] :2156

MediaPresentationImpl::check\_if\_source\_video\_updated:1284:[0x1631a60]: Video source settings - width x height: 1920x1080, clean\_apperture: 1920x1080, pixel\_aspect\_ratio: 1x1, picture\_aspect\_ratio: 1.777778

[INF] 01.12.2016 08:36:18.926 [DVBS\_VIDEO] [615:898 MediaPresentationImpl] :2157

VideoDisplay::get\_aspect\_ratio\_impl:1542:[0x10782e8]: display\_type: 'HD', aspect\_ratio: "16:9"

[INF] 01.12.2016 08:36:18.926 [DVBS\_VIDEO] [615:898 MediaPresentationImpl] :2158

VideoDisplay::match\_dvbs\_aspect\_F] 01.12.2016 08:36:18.927 [DVBS\_VIDEO] [615:898

MediaPresentationImpl] :2161 VideoDisplay::update\_hardware\_settings:1697:[0x10782e8]: called hd content mode

[INF] 01.12.2016 08:36:18.927 [DVBS\_PLATFORM] [615:898 MediaPresentationImpl] :2162

DisplayUpdateMode::set\_mode:45:[0x10786b8]: mode: Mode\_Now

[INF] 01.12.2016 08:36:18.933 [DVBS\_VIDEO] [615:898 MediaPresentationImpl] :2163

VideoDisplay::update\_hardware\_settings:1697:[0x10782e8]: called window settings update

[INF] 01.12.2016 08:36:18.933 [DVBS\_PLAT] 2166

MediaPresentationImpl::switch\_internal\_state:3460:[0x1631a60]: switch internal state, media\_presentation: 1 state: WaitingFrame -> PlayingLive

Рисунок П.13 – Текстовое описание шаблона поведения компонентов ТВР при ошибке "черный экран"

На рисунке П.15 показан шаблон, отражающий характерное поведение компонентов ТВР при успешной записи программ DVR. На рисунке П.15 наблюдаются два пика, на которые указывают стрелки. Пики соответствуют времени записи DVR программ. Временная шкала отложена по оси абсцисс, число записей различных типов показано по оси ординат.

В результате анализа лог файлов, зарегистрированных при различных условиях, сформировано несколько сотен шаблонов.

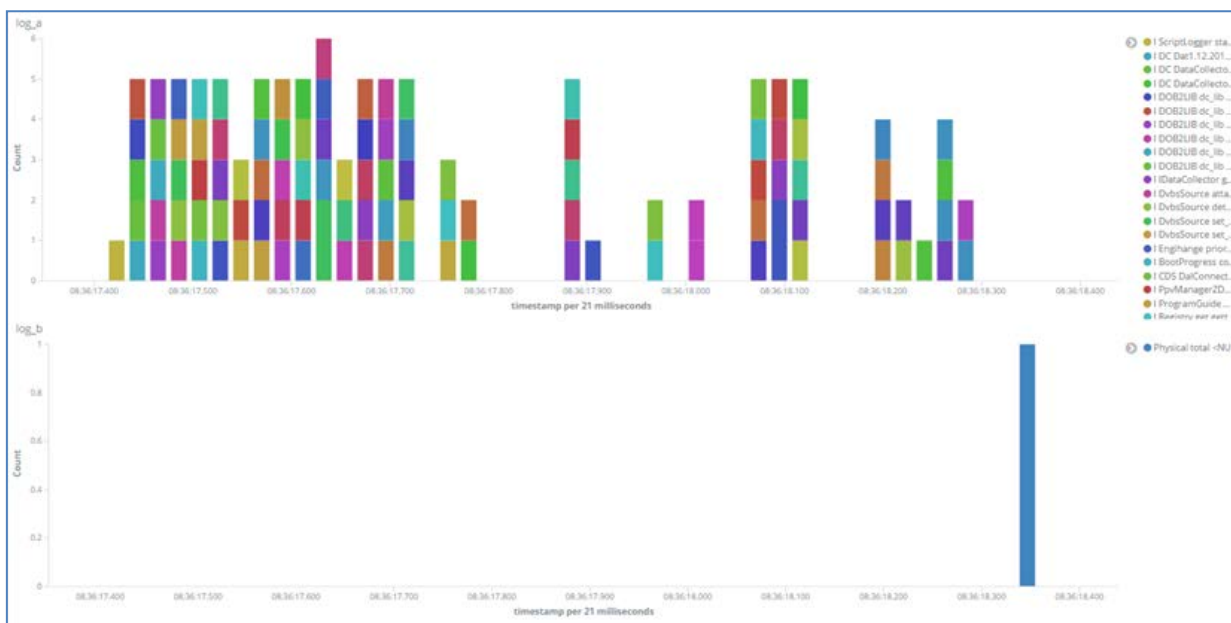


Рисунок П.14 – Графическое представление шаблона поведения программных компонентов ТВР при ошибке "черный экран"

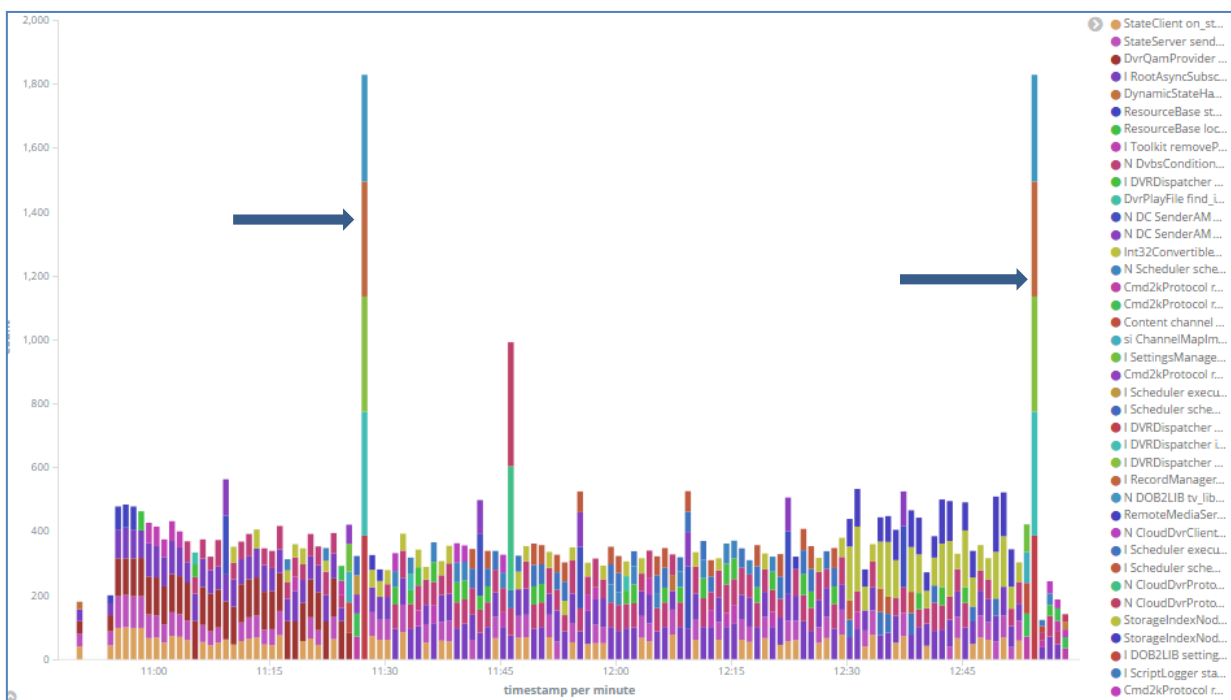


Рисунок. П.15 – Шаблон поведения компонентов ТВР по данным лог файлов при успешной записи программ DVR

Таким образом, была построена модель сети СКЦТ, основу которой составили выявленные зависимости в состоянии параметров различных элементов сети

#### П.4. Результаты построения моделей процессов мониторинга в области телекоммуникаций

Задачи оперативного синтеза процессов и программ решаются при определении регулярных процессов мониторинга, которые целесообразно выполнять на стороне ТВР, а также при формировании специализированных процессов для локализации возникающих

ошибочных ситуаций, выявления причин их возникновения и определении возможных путей устранения.

Процессы и программы мониторинга генерировались сервером, выполняться они могли как на стороне ТВР, так и на стороне сервера.

Регулярный мониторинг предусматривал оперативное выявление и обработку только наиболее часто встречающихся ошибок. К таким ошибкам относятся ошибка "Черный экран" (пользователь видит только черный экран, возможно с каким-то сообщением, обычно это означает проблемы с воспроизведением потока, передаваемого на некоторой частоте), ошибка AVN (отсутствие данных с AVN сервера (AVN сервер присылает параметры для установки AVN сессии, сервер может прислать ошибку, или не ответить на запрос)), ошибка EPG (не был получен или устарел файл с данными о программе передач), ошибка SDV (нерабочее состояние SDV каналов, SDV сервер не доступен или нет частот сервера на миникарусели), ошибка DVR (например, проблемы с диском). В соответствии со статистическими данными эти ошибки составляют порядка 95% всех ошибок, оказывающих негативное влияние на пользовательские функции ТВР.

Одна и та же ошибка, в зависимости от условий, может быть вызвана различными причинами, может оказывать различное влияние на пользователей, так же, как правило, отличаются и пути ее устранения.

В таблице П.8 приведены некоторые из типовых ошибок, возможные причины их возникновения, связанные с ними параметры состояния ТВР, а также информационные сообщения, отображаемые у конечных пользователей на экранах ТВ при их возникновении. Описание параметров состояния ТВР приведено в таблице П.9.

Существенный интерес представляет мониторинг перезагрузок ТВР. Возможные причины перезагрузок определены в таблице П.10. Обработка таких ситуаций предусматривает анализ лог файлов.

Таблица П.8 – Типовые ошибки работы ТВР

Alert Type	Reason Code	ReasonName	ReasonDescription	Requested Parameters 1 step	Requested Parameters 2 step	Requested Parameters 3 step	User context popup
BlackScreen	0	Reason Unknown		n/a			
BlackScreen	32779	BS_TuningTimeout	<p>1. Lineup failure:</p> <p>a) channel's source_id doesn't exist</p> <p>b) incorrect sdv attribute. SDV channels need SDV server to get tuning parameters. If SDV channel in Lineup has got no SDV attribute, STB doesn't attempt to connect server to get these parameters and tries to tune to it using source_id only. see 1.b.iii.</p> <p>c) Vica versa, if a watchtv channel has SDV attribute, STB connects SDV server, requests frequency, modulation, MPEG stream number, but the server can't provide this data.</p> <p>d) incorrect avn attribute. See ii. (this case can't be covered by "no avn" alert, as STB will not recognize this error as Lineup-specific case)</p> <p>Note: At the moment of writing, STB detects this kind of failures as ordinary timed_out errors.</p> <p>2. No first frame:</p> <p>a) PAT is unavailable (PID #0)</p>	<p>channel number, source_id, channel_map id (vct_id)</p> <p>channel number, source_id, channel_map id (vct_id)</p> <p>node_id(Service group ID) hub_id channel_map id (vct_id)</p>	<p>vm_ia_firstQamEqGain</p> <p>vm_ia_firstQamModulationErrorRatio</p> <p>vm_ia_firstQamSignalLevel</p> <p>vm_ia_firstQamSignalToNoiseRatio</p> <p>vm_ia_firstQamCorrected</p> <p>vm_ia_firstQamUncorrected</p> <p>vm_ia_firstQamFrequencyTuned</p> <p>vm_ia_firstQamVideoPids</p> <p>vm_ia_firstQamAudioPids</p>		"Channelunavailable" popup. Reasons" Отсутствует отдельный тип всплывающих пользовательских сообщений для ошибок, возникающих по причинам отсутствия SDV, AVN. Пользователю выводится одинаковое сообщение вне зависимости от причин возникновения ошибки. Текст сообщения зависит не от причины возникшей ошибки, а от действий пользователя.

Alert Type	Reason Code	ReasonName	ReasonDescription	Requested Parameters 1 step	Requested Parameters 2 step	Requested Parameters 3 step	User context popup
				SDV DCM SNR PAT/PMT			
			b) PAT is available, but "mpeg program number" is not specified in it	node_id(Service group ID) hub_id channel_map id (vct_id)			
			c) PAT is available, but PID number (PMT) is not available in the stream Note: These errors can be detected with the help of tuning system, which notifies STB about the reason of failure. In this case OMM agent sends OEM API error received in case of tuning. List of available ARRIS OEM API errors can be found at the bottom of this page.	node_id(Service group ID) hub_id channel_map id (vct_id) SDV DCM SNR PAT/PMT			
			3. Video format is not supported no MPEG4 codec				
BlackScreen	4	BlackScreenReason_NotAuthorized	При попытке подключения к каналу, CAS проверяет его статус методом check_sources_id (платформенная часть PowerUp которая обращается кDVBS), и если скрипту возвращаетсяAuthorizationStatus_Error, то возникает соответствующее пользовательское.	channel number, source_id, entitlements status = "notauthorized" Дополнительные параметры, как правило, не запрашиваются. Канал либо авторизован, либо нет. В некоторых случаях можно проверить работу CAS	stbSumCASRestartCount – stbCASNetworkStatus –	Другие параметры, связанные с CAS в разделе chrCasDialog	<b>"InstantUpgrade" popup</b>  Пользователь переключился на канал, который он не покупал. Ему выводится сообщение с предложением купитьканал.  Точно такое же сообщение будет выведено, если пользователь выбрал

Alert Type	Reason Code	ReasonName	ReasonDescription	Requested Parameters 1 step	Requested Parameters 2 step	Requested Parameters 3 step	User context popup
							переход на канал через Проводник. Но, тогда не возникнет ошибка BlackScreen.
BlackScreen	1	BlackScreenReason_ParentalPin	На экране пользователя появился запрос на ввод пин кода. Если пользователь не ввел пин код, то оверлей исчез. Остался только черный экран. Т.е. причина – отсутствие активности пользователя.				<b>"EnterPIN" popup.</b> Пользователь не ввел пин код.
BlackScreen	2	BlackScreenReason_DecoderState	DecoderState reason is sent when zero-frames are detected.	stbBSECurrentTotal	stbChannelNumberSourceID	Logs of platform lib	
Local DVR Error	0	Reason Unknown					
Local DVR Error	81	UnableToCancel					
Local DVR Error	82	UnableToSchedule		recording parameters, AMS VIP, stbDvrFailures	stbDvrHddRWEError	stbDvrHddTable / Logs of platform lib	
					stbDvrHddNotFound	stbDvrHddPartition	
					stbDvrDriveNumber		
					stbEPGStatus	Logs of DTS lib / Logs of AMS	
Local DVR Error	83	UnableToSave		recording parameters, AMS VIP			

Alert Type	Reason Code	ReasonName	ReasonDescription	Requested Parameters 1 step	Requested Parameters 2 step	Requested Parameters 3 step	User context popup
Local DVR Error	84	UnableToDelete		recording parameters, AMS VIP			
Local DVR Error	201	NoSpaceRecordingFailure	There is no room for storing records.	stbESataAuthorized	stbDvrHddTable	Logs of platform lib	
				stbDvrHddNotFound	stbDvrHddPartition		
				stbDvrDriveNumber stbDvrHddRWEror stbDvrFailures			
AVN Error	0 (180)	AVN-SRV-NA	AVN session setup fails due to AVN server unavailability (due to connection error or some other reason)	Logs of NS lib	Logs of platform lib		
AVN Error	1 (181)	AVN-APP-NA	AVN session setup fails due to application (AVN service) unavailability				
AVN Error	2 (183)	AVN-SRVRES-NA	AVN session setup fails due to inability to allocate resource on AVN serve				
AVN Error	3 (185)	AVN-NO-TUNEINFO	AVN session setup fails if Session_Setup_Response received without Tuning Info parameters				
AVN Error	4 (186)	AVN-INV-PAR	AVN session setup fails if session_setup returns an Invalid parameter				
AVN Error	5 (187)	AVN-TUNE-FAIL	The client is not able to tune to specified freq.pn,qam; AVN session setup fails	stbBSECurrentTotal	stbChannelNumberSourceID	Logs of platform lib	
					chrSTBEASDiagP	Logs of platform lib	
					stbTSIDFreq		
AVN Error	6 (188)	AVN-TSID-MISM	AVN session setup fails due to TSID mismatch				

Alert Type	Reason Code	ReasonName	ReasonDescription	Requested Parameters 1 step	Requested Parameters 2 step	Requested Parameters 3 step	User context popup
AVN Error	7 (182)	AVN-SETUP-TIME	AVN session setup fails due to AVN connection timeout reached				
AVN Error	8 (192)	AVN-SRV-TIME	Running AVN session fails when AVN returns the error code that indicates inactivity timeout reached				
AVN Error	9 (189)	AVN-STRM-LOST	The client determines that there is no longer the stream from the server and closes the running AVN session				
AVN Error	10 (190)	AVN-CL-HB	The client determines that heartbeat from AVN server was lost and closes the running AVN session				
AVN Error	11 (191)	AVN-SRV-ERR	Running AVN session fails when AVN returns server error				
AVN Error	12 (193)	AVN-SRV-HB	Running AVN session fails when AVN returns the error code that indicates heartbeat from the client was lost				
AVN Error	13 (195)	AVN-CL-INTL	The client closes the running/initiated AVN session due to an internal client error				
AVN Error	14 (196)	AVN-MSG-FAIL	The client closes the running/initiated AVN session when sending any message to the server failed				
AVN Error	15 (197)	AVN-NET-BAD	Network instability. The client determines that the network is in a state that could no longer provide				
AVN Error	17 (194)	AVN-SRV-QUAR	Running AVN session fails when AVN returns the error code that indicates quarantine				
AVN Error	18 (184)	AVN-SRV-SETUPERR	AVN session setup fails due to another AVN server error				
No EPG Error	0	UNKNOWN					
No EPG Error	1	FileNotFound	The file does not exist.	file name, URLstbEPGStatus	Logs of DTS lib	Logs of AMS	
				file name, URLstbEPGLastUpdate	Profile		
No EPG Error	2	FileLoadingError	Impossible to load the file	file name, URL			



Alert Type	Reason Code	ReasonName	ReasonDescription	Requested Parameters 1 step	Requested Parameters 2 step	Requested Parameters 3 step	User context popup
No EPG Error	3	InvalidVersion	An invalid file version was passed	file name, URL			
No EPG Error	4	NotEnoughMemory	Insufficient memory	file name, URL			
No EPG Error	5	LoadingTimeout	A timeout occurs during file loading	file name, URL			
No EPG Error	6	ParseError	Impossible to parse the loaded data	file name, URL			
No EPG Error	13	URL_Unknown		file name, URL			
SDV Error	26	SDV_NoMiniCarousel	No minicarousel location was provided				
SDV Error	27	SDV_TuningError	Tuner/filter error				
SDV Error	28	SDV_MC_ReadingTimeout	Minicarousel reading timeout				
SDV Error	29	SDV_MC_ParsingError	Microcarousel Parsing error				

Таблица П.9 – Параметры состояния ТВР

Параметр	Описание
BlackScreen	
Базовые платформенно независимые параметры	
Channelnumber	Номер канала
source_id	Технический идентификатор канала, передаваемый тюнеру для настройки на выбранный канал
channel_map id (lineup_id / vct_id / hub_id)	Идентификатор сетки каналов и параметры, необходимые для его получения. Если канал с необходимым source_id найден, то к такому каналу можно подключиться. Если канал отсутствует, то возникает ошибка.
SDVDCM	Динамическая сетка каналов, которые вещаются с помощью технологии Switched Digital Video
SNR (Signal / Noise Ratio)	Отношение сигнал/шум, определяющее качество сигнала. Это очень низкоуровневый параметр, характеризующий наличие проблем качеством сигнала в сети.
PAT/PMT for private source ID	Program Allocation Table / Program Map Table для source_id определенного канала. Представляет собой таблицу программ (каналов) и их характеристик, без них невозможно определить на какой канал выполняется переключение. По нему определяется куда подключаться для получения потока данных от AVN сервера
Service group ID (SG Number)	Номер сервисной группы для данного устройства
Параметры, связанные с различными ошибками	
stbBSECurrentTotal	Параметр, показывающий количество ошибок декодера при выводе видео. Позволяет предварительно определить есть ли проблема с выводом видео
stbChannelNumberSourceID	Номер канала и source_id
chrSTBEASDiagP	Параметры показывающие работу EAS, помогают определить не был ли тюнер отобран EAS
stbTSIDFreq	TS_ID и частота текущего канала
Local DVR parameters	
stbDvrDriveNumber	Номер диска используемого DVR
stbDvrFailures	Счетчик ошибок жесткого диска при работе с DVR
stbDvrHddNotFound	Указывает на отсутствие доступа к HDD, со стороны DVR
stbDvrHddPartition	Таблица разделов жестких дисков
stbDvrHddRWError	Количество ошибок чтения/записи HDD
stbDvrHddTable	Таблица с информацией о переходе дисков в spindown
stbEPGLastUpdate	Последнее обновление EPG данных
stbEPGStatus	Статус EPG: Loaded, Failed, Expired, Empty
stbESataAuthorized	Показывает авторизован ли ТВР для использования внешнего диска
Платформенно-зависимые параметры тюнера	
vm_ia_firstQamEqGain	Показываетуровень сигнала QAM (Quadrature Amplitude Modulation) тюнера
vm_ia_firstQamModulationErrorRatio	Показывает ошибки qam тюнера
vm_ia_firstQamSignalLevel	Показываетуровень QAM сигнала
vm_ia_firstQamSignalToNoiseRatio	Показывает соотношение уровня сигнала к шуму
vm_ia_firstQamCorrected	Показывает количество of исправленных ошибокQAM
vm_ia_firstQamUnCorrected	Показывает количество неисправленных ошибок QAM
vm_ia_firstQamFrequencyTuned	Показывает the частоту на которую тюнер настроен
vm_ia_firstQamVideoPids	Показывает список видео PIDs в потоке

Параметр	Описание
vm_ia_firstQamAudioPids	Показывает список аудио PIDs в потоке
Параметры CAS	
stbSumCASRestartCount	Количество перезапусков CAS
stbCASNetworkStatus	Функционирование сети CAS. Да/Нет (1/0)

Таблица П.10 – Причины перезагрузок ТВР

Код причины перезагрузки	Причина перезагрузки
0	NULL,
1	UNKNOWN
2	SIGTRAP
3	DUMP_REQUESTED
4	SIGTERM
5	SIGHUP
6	SIGINT
7	SIGWINCH
8	SIGFPE
9	SIGABRT
10	SIGILL
11	SIGUSR1
12	SIGBUS
13	SIGQUIT
14	SIGSEGV

Программы, синтезированные для регулярного мониторинга, выполнялись на стороне ТВР. Фрагмент одной из программ показан на рисунке П.16. Представленная в примере программа включает процедуру регистрации процесса регулярного мониторинга на ТВР и процедуры выявления следующих ошибок: ошибка SDV, ошибка EPG, ошибка "черный экран", отсутствие данных с AVN.

```

start()
{
// регистрация программы
register("dc:AppLaunch(int screen, int startTime, int timestampPartMsec, int reason, int channel)",
AppLaunch);
    register("dc:AppSessionComplete(int app, int appEndTime, int timestampPartMsec, int sessionId, int
reason, int clientSessionId)", AppSessionComplete);
// регистрация контролируемых ошибочных ситуаций
    register("dc:EpgDataError(int date,int type,int error)", EpgDataError);
    register("dc:BlackScreen(int reason)", BlackScreen);
    register("dc:SdvError(int reason)", SdvError);
// сбор отдельных параметров о ТВР
reportInt(_hubId);
reportInt(_nodeId);
    reportStr(_uiFlashVersion);
    reportStr(_uiRomVersion);
reportStr(_uiBuildNum);
}

```

```

// обработка данных об ошибочной ситуации, соответствующей ошибке отсутствия svd
SdvError(params)
{
    if (SendAlertCommon(SDV_ERROR_EVENT))
        reportInt(intArg());
}
// обработка данных об ошибочной ситуации, соответствующей ошибке отсутствия программы
// передач
EpgDataError(params)
{
    if (SendAlertCommon(NOEPG_EVENT))
    {
        fileDate = intArg();
        srcType = intArg();
        reason = intArg();
        reportInt(srcType);
        reportInt(fileDate);
        reportInt(reason);
    }
}
// обработка данных об ошибочной ситуации, соответствующей отсутствию изображения
BlackScreen(params)
{
    SendBlackScreen(-1, intArg());
}
// обработка данных об ошибочной ситуации, соответствующей отсутствию соединения с AVN
// сервером
AppSessionComplete(params)
{
    app = intArg();
    end_time = intArg();
    timestampPartMsec = intArg();
    session_id = intArg();
    reason = intArg();
    clientSessionId = intArg();

    if (app == AVN_APPLICATION && reason >= AVN_SERVER_ERROR_REASON &&
        SendAlertCommon(NOAVN_EVENT))
    {
        reportInt(_avnServerIp);
        reportInt(_avnServerPort);
        reportInt(reason - AVN_SERVER_ERROR_REASON); // error
    }
}

```

Рисунок П.16. Фрагмент синтезированной программы для регулярного мониторинга

Для локализации ошибок формировались свои программы в зависимости от типов ошибок. Программы исполнялись как на стороне ТВР, так и на стороне сервера. Ниже рассмотрены процессы и программы мониторинга, сформированные для локализации ошибок "черный экран".

Для ошибок типа "черный экран", с учетом объема требуемой обработки и объема передаваемых данных, непосредственно на ТВР обрабатывались ошибки, возникающие по причинам превышения допустимого времени ожидания (BS\_TuningTimeout), проблем

переключения на другой канал (BlackScreenReason\_NotAuthorized), проблем с авторизацией (BlackScreenReason\_ParentalPin). Ошибки, связанные с состоянием тюнера, (DecoderState) обрабатывались на стороне сервера. Их локализация и устранение требовали затрат существенных ресурсов, что определялось необходимостью получения лог файлов и их обработки.

Процедура локализации и устранения ошибок являлась многоуровневой. Она предусматривала многократное последовательное уточнение параметров состояния компонентов СКЦТ исходя из значений ранее запрошенных параметров.

На рисунке П.17 показана сформированная для ТВР программа, которая использовалась для выявления предварительных причин возникновения ошибки "черный экран". Установленная на стороне ТВР предварительная причина передавалась на сервер.

*/\* идентификация ошибочной ситуации отсутствия видео \*/*

*BlackScreenReason (screen, reason)*

```

{
    no_video_reason =
        screen == -1 ? reason :
        screen == TUNING_FAILED_SCREEN && reason == APP_LAUNCH_REASON ?
BLACKSCREEN_TUNING_FAILED :
        screen == INSTANT_UPGRADE_SCREEN && reason == APP_LAUNCH_REASON ?
BLACKSCREEN_NOT_AUTHORIZED :
        screen == PARENTAL_PIN_PROMT_SCREEN && reason == APP_LAUNCH_REASON
? BLACKSCREEN_PARENTAL_PIN :
        -1;

    if (blackscreen_reason == -1)
        return false;
    if (SendAlertCommon(BLACKSCREEN_EVENT))
        reportInt(blackscreen_reason);
    return true;
}

```

Рисунок П.17 – Фрагмент сформированной программы для определения предварительных причин возникновения ошибки "черный экран"

Так, при получении сервером информации об ошибке "черный экран" по причине BS\_TuningTimeout, формировалась программа для получения значений параметров отдельных компонентов ТВР. При этом использовалась информация о связи ошибки "черный экран" и состояния компонентов системы тюнинга. Эта информация позволяла сузить область анализируемых параметров до параметров одной системы. К информативным параметрам системы тюнинга с точки зрения локализации и устранения ошибки "черный экран" относятся параметры, представленные в таблице П.9. Состав параметров и их общее число зависит от типа ТВР. Обычно анализируется несколько десятков параметров.

Программы, сформированные на сервере для получения значений общих и специализированных параметров системы тюнинга при возникновении ошибки "черный экран", приведены на рисунке П.18 и рисунке П.19. Эти программы передавались на ТВР, где осуществлялся сбор требуемых параметров.

```
/* сбор параметров о состоянии STB для локализации проблемной ситуации */
SendNoVideoDiag(params)
{
    reportInt(vm_ia_source_id)

    SDV DCM
    SNR (Signal / Noise Ratio)
    PAT/PMT for private source ID
    SGNumber
    reportInt(vm_ia_lineup_id);
    reportInt(vm_ia_SDV DCM);
    reportInt(vm_ia_SNR);
    reportInt(vm_ia_PAT_PMT);
    reportInt(vm_ia_SG Number);
}
```

Рисунок П.18 – Программа, сформированная для получения значений общих параметров при возникновении ошибки отсутствия изображения

Далее запрашивались специализированные параметры системы тюнинга.

```
/* сбор параметров о состоянии STB для локализации проблемной ситуации */
SendNoVideoDiag(params)
{
    reportInt(vm_ia_firstQamEqGain);
    reportInt(vm_ia_firstQamModulationErrorRatio);
    reportInt(vm_ia_firstQamSignalLevel);
    reportInt(vm_ia_firstQamSignalToNoiseRatio);
    reportInt(vm_ia_firstQamCorrected);
    reportInt(vm_ia_firstQamUnCorrected);
    reportInt(vm_ia_firstQamFrequencyTuned);
    reportInt(vm_ia_firstQamVideoPids);
    reportInt(vm_ia_firstQamAudioPids);
}
```

Рисунок П.19 – Программа, сформированная для получения значений специальных параметров системы тюнинга при возникновении ошибки отсутствия изображения

На основе данных о значениях параметров, полученных от ТВР, уточнялась причина ошибки. По уточненным данным ошибка могла быть устранена. При необходимости проведения дальнейшего более глубокого анализа причины осуществлялся сбор и анализ дополнительной информации, например, содержащейся в лог файлах ТВР.

В ситуациях, когда вероятной причиной ошибки "черный экран" являлась неисправность декодера, формировались программы, выполняемые на стороне сервера. На

рисунке П.20. показана общая логика обработки таких ошибок, на рисунке П.21 - алгоритм работы сервера. Приведенный алгоритм определяет порядок обработки связанных параметров состояния различных элементов сети. В рассматриваемом случае он также предусматривает получение и обработку лог файлов ТВР.

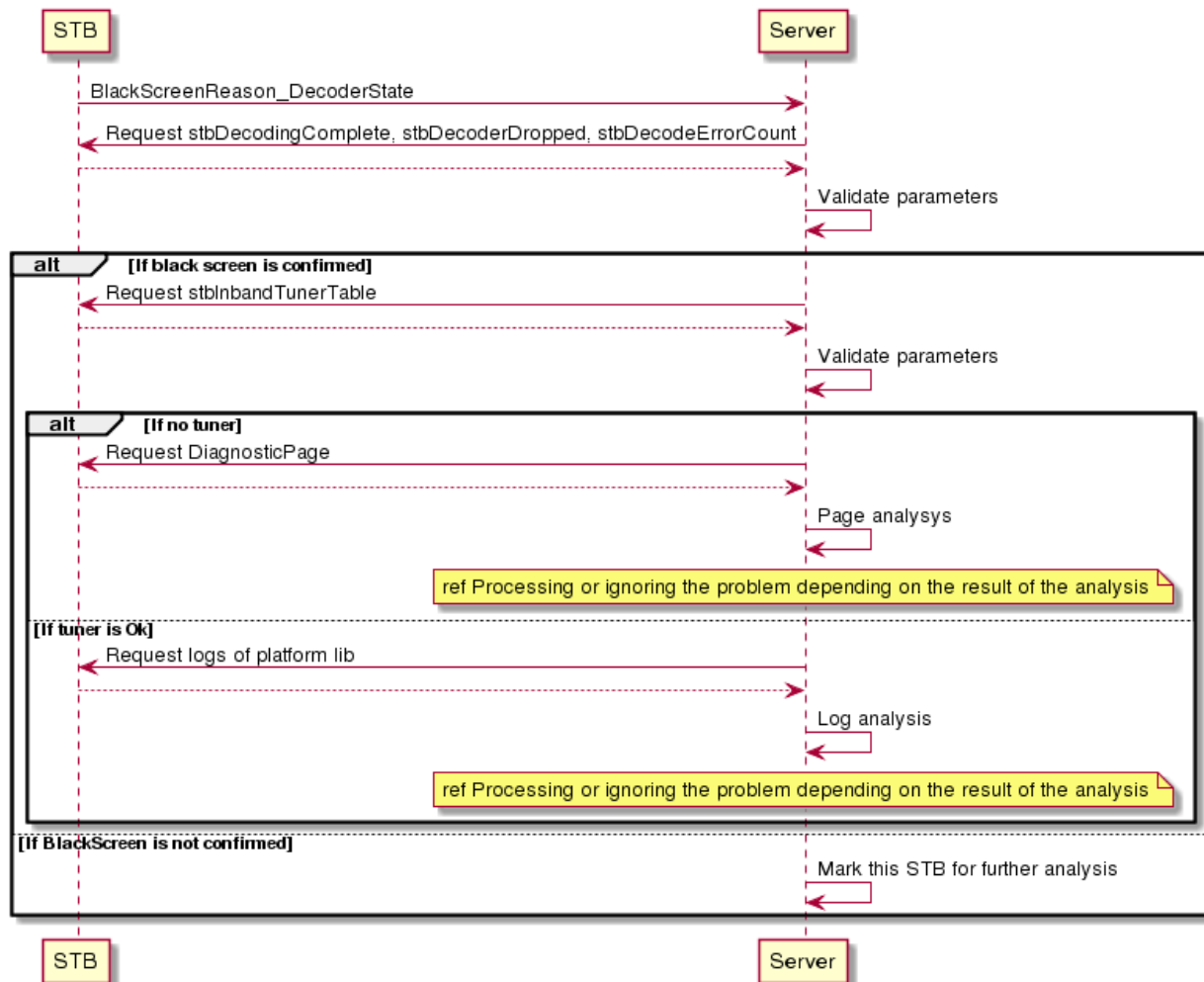


Рисунок П.20 – Логика обработки ошибки "черный экран" по причине неисправности декодера

BlackScreenReason\_DecoderState

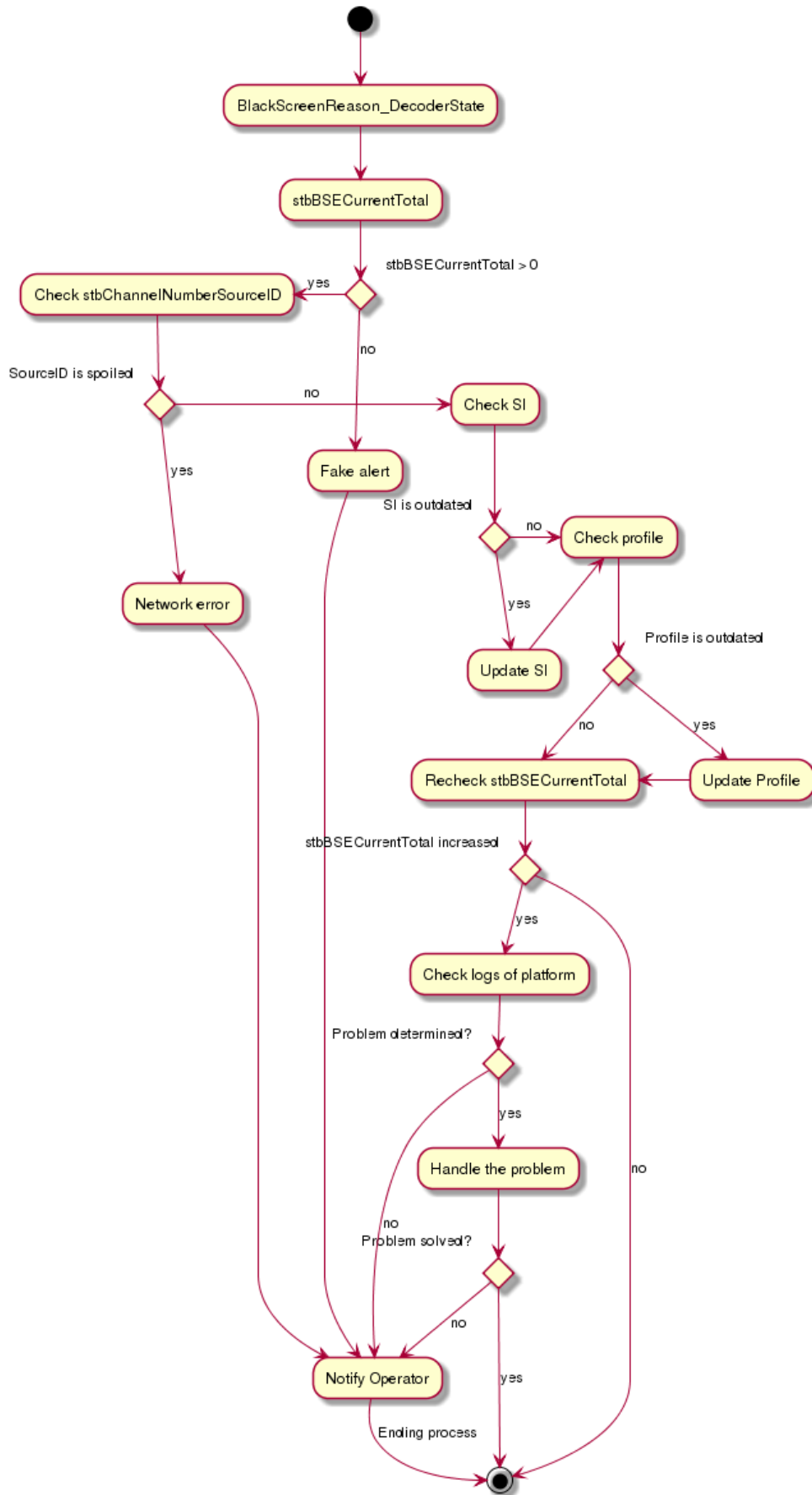


Рисунок П.21 – Алгоритм обработки ошибки "черный экран" по причине неисправности декодера



В соответствии с алгоритмом была сформирована программа обработки ошибок по причине неисправности декодера (рисунок П.22):

```

/*обработка ошибок "черный экран" по причине неисправности декодера*/
ifreason == "BlackScreenReason_DecoderState"
StbParamRequest ("stbBSECurrentTotal")
  if "stbBSECurrentTotal" > 0 then
    NotifyMessage ("Alert is not confirmed")
  else
    StbParamRequest ("stbChannelNumberSourceID")
    ActiveErrorsTableFind(SourceID)
      if true then
        NotifyMessage (SourceID, " is spoiled")
      else
        StbParamRequest (SiStatus)
          if SiStatus == outdated then
            DoUpdateSi()
          else
            StbParamRequest (ProfileStatus)
              if ProfileStatus == outdated then
                DoUpdateProfile()
            StbParamRequest ("stbBSECurrentTotal")
              if stbBSECurrentTotal > stbBSECurrentTotalOld then
                StbLogsRequest (platform)
                DoLogAnalysis ()
                DoHandleResult ()
                NotifyMessage (Result)

```

Рисунок П.22 – Синтезированная программа обработки ошибочной ситуации "черный экран" на стороне сервера

Локализацию ошибок за счет обработки лог файлов рассмотрим на примере ошибки в работе пользовательского сервиса Pay-Per-View (PPV). Этот сервис обеспечивает возможность покупки фильмов, передач и т.д. Ниже рассматриваются записи в лог файле, свидетельствующие о возникновении ошибки и позволяющие ее локализовать.

При запуске PPV сервис обмена данными Data Transfer Service (DTS) получает запрос на открытие файла, о чем свидетельствует запись в лог файле:

```

[INF] 30.03.2017 11:27:04.398 [DataTransportService] [905:953 common.scheduler.thread] :155377
[I] DataTransportService: name=ppv/oob/ppv-2017033000.zdb, version=0, async_id=1515212830,
callback=0x505f04

```

По данным того же лог файла (несколькими строками ниже) указано, что загрузка файла была прервана в силу проблем с конфигурацией ТВР:

```

[ERR] 30.03.2017 11:27:04.399 [DataTransportService] [905:969 dts.scheduler.thread] :155380 [H][E]
TransportSystem: get_data: can't found appropriate transport for data ppv/oob/ppv-2017033000.zdb!!!

```

Об этом факте было извещено приложение. О чем свидетельствует строка:

[INF] 30.03.2017 11:27:04.400 [DataTransportService] [905:969 dts.scheduler.thread] :155384 [I] DirectFileSystem: on\_data\_error\_sync: unable to fetch ppv/oob/ppv-2017033000.zdb in 1 attempts

После выявления системой мониторинга ошибки в конфигурации ТВР могут быть предприняты действия по ее устранению. Возможны такие альтернативные варианты: отправка уведомления оператору, загрузка необходимого файла из кэш, задействование других видов транспорта для передачи файла.

Другими примерами ошибочных ситуаций, которые могут быть выявлены по файлам логов, являются: выявление ошибок загрузки данных, ошибок целостности данных, определение причин отмены запроса или повторных циклических перезапросов.

Для устранения этих ошибочных ситуаций, в зависимости от восстановленного сценария их развития, могут быть предприняты меры, включающие инвалидацию кэш, перезапуск сервиса DTS, переконфигурацию прикладных сервисов, по запросу которых выполнялась загрузка файлов и другие.

## **П.5. Оценка эффективности построения моделей наблюдаемых объектов и процессов мониторинга в области телекоммуникаций**

Эффективность построения моделей наблюдаемых объектов и процессов мониторинга оценивалась по следующим ключевым показателям:

- число одновременно наблюдаемых объектов;
- число своевременно выявленных и локализованных ошибок.

Число одновременно наблюдаемых объектов определяется производительностью систем мониторинга. Число своевременно выявленных и устраненных ошибок - полнотой синтезируемых моделей объектов.

*Оценка производительности ППСМ СКЦТ.* Для оценки производительности ППСМ определялась допустимая нагрузка на компоненты сбора (ОММ Listener) и обработки данных (ОММ Handler) мониторинга в ППСМ СКЦТ. Сценарии оценки допустимой нагрузки на эти компоненты приведены в таблице П.11.

Таблица П.11 – Сценарии оценки допустимой нагрузки на компоненты сбора и обработки данных в ППСМ СКЦТ

Сценарий	Серверный компонент	Описание сценариев
Сценарий 1 (нагрузочный сценарий)	ОММ Listener	Поток сообщений об ошибках от ТВР, начиная с 1000 сообщений/с
Сценарий 2	ОММ Handler	Поток сообщений об ошибках от ТВР, начиная с 300 сообщений/с
Сценарий 2-1 (нагрузочный сценарий)	ОММ Handler	Масштабируемость ОММ Handler
Сценарий 2-2	ОММ Handler	Поток сообщений об ошибках от ТВР,

Сценарий	Серверный компонент	Описание сценариев
(сценарий длительной работы)		начиная с 300 сообщений/с
Сценарий 3 (нагрузочный сценарий)	AMS Logs Handler	Сбор лог файлов от ТВР, начиная с 300 сообщений/с, 0.1% - сообщения об ошибках

**Сценарий 1.** Оценка производительности компонента сбора данных (ОММ Listener).

Цель – оценка максимального числа сообщений об ошибках, которые способен получать ОММ Listener и возможность масштабирования подсистемы сбора данных в ППСМ СКЦТ. Сценарий предусматривал генерацию сообщений об ошибках и их отправку компоненту ОММ Listener. Генератором имитировалось поступление сообщений более, чем от 500 различных ТВР. Выполнение сценария предусматривало увеличение / снижение скорости генерации сообщений об ошибках на 50 сообщений/с. В результате было установлено максимально допустимое число сообщений, которое может быть получено одним компонентом ОММ Listener. Критерий оценки успешности приема сообщений компонентом – стабильное среднее время ответа компонента. Время ответа не должно было увеличиваться минимум в течение пяти минут.

*В результате выполнения сценария установлено, что стабильная работа компонента ОММ Listener наблюдается при числе сообщений, не превышающих 600 сообщений/с.* Также было установлено, что число разворачиваемых компонентов ОММ Listener практически не оказывает влияния на пропускную способность при следующих условиях:

- используется gdr транспорт, т.е. доставка сообщений осуществляется с подтверждением;
- сервера образуют кластер серверов, кластерные решения строятся на основе технологий Hazelcast.

Такое поведение объясняется тем, что существенные ресурсы тратятся на сбор элементов, относящихся к одному информационному пакету. К таким элементам относятся само сообщение, подтверждение получения сообщений и другие.

В случае, если используется, например, udr транспорт, возможно масштабирование за счет увеличения числа ОММ Listener. При увеличении числа компонентов сбора данных наблюдается увеличение производительности близкое к линейному.

**Сценарий 2.** Оценка производительности компонента обработки поступающих данных (ОММ Handler).

В соответствии со сценарием оценивались:

а) максимальное число сообщений об ошибках, которые могут быть обработаны компонентом ОММ Handler;

б) возможность масштабирования подсистемы обработки сообщений об ошибках за счет увеличения числа компонентов ОММ Handler.

При генерации сообщений имитировалось поступление сообщений об ошибках более, чем от 500 различных ТВР. Начальная частота генерации сообщений составляла 300 сообщений/с. В процессе выполнения сценария число генерируемых сообщений увеличивалось и уменьшалось на 50 сообщений/с в зависимости от состояния ОММ Handler. В результате было установлено максимально допустимое число сообщений, которое может быть обработано одним компонентом ОММ Handler. Критерий оценки успешности обработки сообщений компонентом ОММ Handler – очередь сообщений, ожидающих обработку компонентом ОММ Handler, не увеличивается минимум в течение пяти минут, все сообщения успешно размещаются в БД ППСМ СКЦТ.

На графиках рисунка П. 23 показаны зависимость длины очереди сообщений от времени работы компонента (верхний график) и зависимости числа полученных (нижний график, желтый цвет) и числа успешно обработанных (нижний график, красный цвет) сообщений от времени работы компонента. Из рисунка (рисунок П. 23, верхний график) видно, что критически очередь растет после приблизительно 15 минут нагрузки в 300 сообщений/с. После выключения нагрузки буфер очереди был полностью обработан. При этом далее (с 15:40 до 9:20) была зафиксирована стабильная работа тестируемой конфигурации при нагрузке в 300 сообщений/с (рисунок П.24). *Таким образом, результаты выполнения сценария с учетом сформулированного критерия успешности его выполнения позволили установить, что нагрузка в 300 сообщений/с является допустимой для компонента ОММ Handler.*



Рисунок П.23 – Состояние компонента обработки данных (ОММ Handler) при загрузке 300 сообщений/с на интервале времени с 14:50 до 15:50: зависимость длины очереди сообщений от времени работы компонента (верхний график); зависимости числа полученных (нижний график, цвет желтый) и обработанных сообщений (нижний график, цвет красный) от времени работы компонента



Рисунок П.24 – Состояние компонента обработки данных (OMM Handler) при загрузке 300 сообщений/ с на интервале времени с 15:40 до 9:20:

зависимость длины очереди сообщений от времени работы компонента (верхний график); зависимости числа полученных (нижний график, цвет желтый) и обработанных сообщений (нижний график, цвет красный) от времени работы компонента

**Сценарий 2-1.** Оценка производительности компонента обработки поступающих данных (OMMHandler) при наличии двух компонентов OMMHandler

Сценарий предусматривал генерацию сообщений об ошибках и их обработку средствами двух компонентов OMM Handler. Условия генерации сообщений были аналогичны условиям, определенным для первого варианта реализации сценария. Отличие состояло в начальной частоте генерации сообщений. Стартовая нагрузка была увеличена и составляла 500 сообщений/с. В результате было определено максимально допустимое число сообщений, которое может быть обработано каждым из компонентов OMM Handler.

В результате выполнения сценария установлено:

- при нагрузке в 500 сообщений / с после 12 минут начали появляться "скачки" в очереди сообщений (рисунок П. 25), впоследствии все сообщения были нормально

обработаны. На рисунке П.25 стрелкой указано начало выполнения сценария. При 500 сообщений/с наблюдалась стабильная работа системы в течение 50 минут и далее;

- при нагрузке в 800 сообщений / с через 12 минут начали наблюдаться "скачки" в очереди, в течение 3х минут все сообщения в очереди обрабатывались, длина очереди уменьшалась до нуля, потом снова наблюдался рост очереди (рисунок П.26). Очередь сохранялась в течение 6 минут, затем все сообщения были успешно обработаны. Далее опять наблюдался рост очереди. Тестирование проводилось в течение 25 минут;

- при нагрузке в 1000 сообщений / с сразу после старта очередь начинала расти (рисунок П.27). При 1000 сообщений в секунду после 8 минут наблюдался критический рост загрузки процессора (рисунок П.28).

*Таким образом, результаты выполнения сценария с учетом сформулированного критерия успешности его выполнения позволили установить, что нагрузка в 800 сообщений/с является допустимой для ППСМ СКЦТ при двух компонентах ОММ Handler.*



Рисунок П.25 – Зависимости длины очереди от времени работы двух ОММ Handler при нагрузке 500 сообщений/с:

зависимость длины очереди сообщений от времени работы компонента (верхний график); зависимость числа полученных (нижний график, цвет желтый) и обработанных сообщений (нижний график, цвет красный) от времени работы компонента



Рисунок П.26 – Зависимость длины очереди от времени работы двух ОММ Handler при нагрузке 800 сообщений/с:

зависимость длины очереди сообщений от времени работы компонента (верхний график); зависимости числа полученных (нижний график, цвет желтый) и обработанных сообщений (нижний график, цвет красный) от времени работы компонента





Рисунок П.27 – Зависимость длины очереди от времени работы двух ОММ Handler при нагрузке 1000 сообщений/с  
зависимость длины очереди сообщений от времени работы компонента (верхний график); зависимости числа полученных (нижний график, цвет желтый) и обработанных сообщений (нижний график, цвет красный) от времени работы компонента

```
top - 18:59:59 up 6:31, 5 users, load average: 1,22, 0,92, 0,82
Tasks: 148 total, 1 running, 147 sleeping, 0 stopped, 0 zombie
top - 20:45:39 up 1 day, 8:16, 5 users, load average: 3,29, 2,50, 2,03
Tasks: 157 total, 2 running, 155 sleeping, 0 stopped, 0 zombie
%Cpu(s): 88,2 us, 2,8 sy, 0,0 ni, 5,9 id, 2,1 wa, 0,0 hi, 1,0 si, 0,0 st
KiB Mem : 7895668 total, 180180 free, 5046072 used, 2669416 buff/cache
KiB Swap: 4984828 total, 4725464 free, 259364 used. 2341196 avail Mem
```

PID	USER	PR	NI	VIRT	RES	SHR	S	%CPU	%MEM	TIME+	COMMAND
23674	root	20	0	5830048	2,144g	7100	S	326,6	28,5	321:06.12	java
30774	root	20	0	5521528	274420	16400	S	8,3	3,5	6:10.36	java
30806	root	20	0	5521528	266064	16352	S	8,3	3,4	6:17.21	java
18662	root	20	0	5521528	438264	7344	S	8,0	5,6	175:46.67	java
18681	postgres	20	0	342072	143684	142104	S	7,6	1,8	18:01.20	postgres
30825	postgres	20	0	342072	144684	142168	S	7,6	1,8	4:57.24	postgres
18696	root	20	0	5521528	455780	7392	S	7,3	5,8	175:40.93	java
18715	postgres	20	0	342072	143496	142076	D	7,3	1,8	18:05.73	postgres
<b>30792</b>	<b>postgres</b>	<b>20</b>	<b>0</b>	<b>342072</b>	<b>144676</b>	<b>142160</b>	<b>R</b>	<b>7,3</b>	<b>1,8</b>	<b>5:02.10</b>	<b>postgres</b>
1	root	20	0	193628	4184	2748	S	0,0	0,1	0:01.83	systemd
2	root	20	0	0	0	0	S	0,0	0,0	0:00.01	kthreadd
3	root	20	0	0	0	0	S	0,0	0,0	0:00.34	ksoftirqd/0
5	root	0	-20	0	0	0	S	0,0	0,0	0:00.00	kworker/0:0H
7	root	rt	0	0	0	0	S	0,0	0,0	0:00.14	migration/0
8	root	20	0	0	0	0	S	0,0	0,0	0:00.00	rcu_bh
9	root	20	0	0	0	0	S	0,0	0,0	4:25.36	rcu_sched
10	root	rt	0	0	0	0	S	0,0	0,0	0:00.98	watchdog/0
11	root	rt	0	0	0	0	S	0,0	0,0	0:01.07	watchdog/1
12	root	rt	0	0	0	0	S	0,0	0,0	0:00.02	migration/1

Рисунок П.28 – Зависимость загрузки процессора от времени работы двух ОММ Handler при нагрузке 1000 сообщений/с

В таблице П.12 приведены сводные результаты выполнения первого и второго сценариев.

Таблица П.12 – Сводная таблица результатов выполнения первого и второго сценариев

Сценарий	Продолжительность (секунды)	Отправлено сообщений	Интенсивность отправки	Ошибок отправки	Компонент	
					Название	Число
Сценарий 1(a)	4667	1400218	300 с/с	0	OMM listener	1
Сценарий 1 (b)	61174	19552411	300 с/с	0	OMM Handler	2
Сценарий 2 (a)	2664	1332270	500 с/с	0	OMM Handler	2
Сценарий 2 (b)	1203	962476	800 с/с	0	OMM Handler	2

**Сценарий 2 - 2.** Оценка устойчивости работы компонента обработки поступающих данных (OMMHandler).

Для оценки устойчивости исследовалась работа OMM Handler на достаточно длительном временном интервале. Компонент стабильно работал в течение 3х дней с нагрузкой в 300 сообщений/с. При этом был зафиксирован рост очереди до 3000-6000 сообщений в течение 3-5 минут с последующей полной обработкой очереди. Такие скачки наблюдались с периодичностью в 8-12 минут.

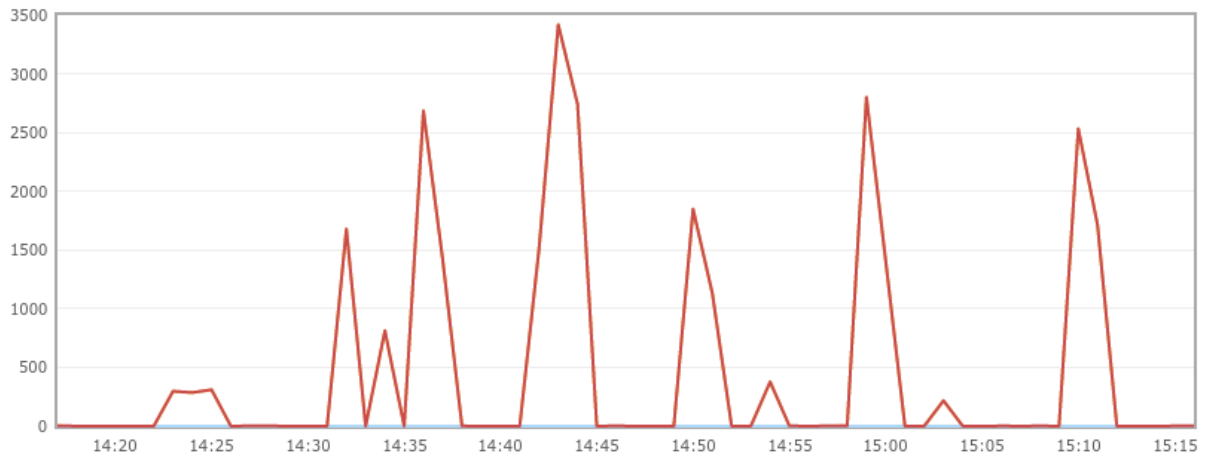
**Сценарий 3.** Оценка производительности компонента обработки лог файлов (AMS Logs Handler).

Сценарий был ориентирован на выявление максимального числа записей лог файлов AMS, которые могут быть обработаны AMS Logs Handler за секунду.

Сценарий предусматривал генерацию 300 записей и их сохранение в виде лог файлов. Для этого использовался специализированный генератор серверных лог файлов. Порядка 0,1% записей лог файлов содержали текстовые подстроки, которые наблюдаются при возникновении ошибочных ситуаций на ТВР. Критерий оценки успешности обработки записей лог файлов компонентом AMS Logs Handler – стабильная работа компонента в течение одного часа.

При нагрузке в 333,3 записи/с в течение часа наблюдалось допустимое поведение (рисунок П.29). В этот период очередь возростала, однако, затем, обрабатывалась. Потеряно небольшое количество сообщений.

Queued messages (chart: last hour) (?)



Message rates (chart: last hour) (?)

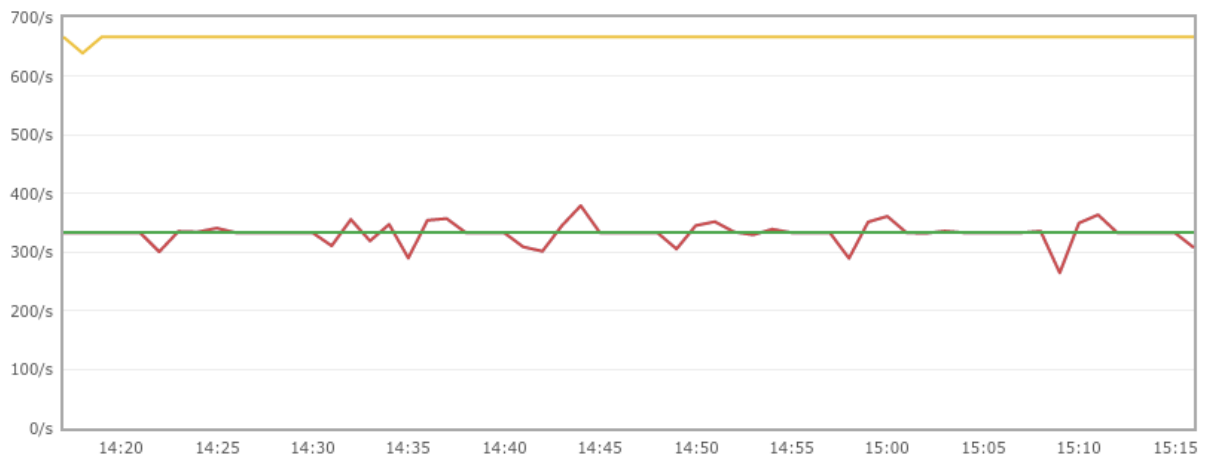


Рисунок П.29 – Зависимость длины очереди от времени работы одного AMS Logs Handler при нагрузке 300,3записей/с

зависимость длины очереди сообщений от времени работы компонента (верхний график); зависимости числа полученных (нижний график, цвет желтый), обработанных (нижний график, цвет красный) и не обработанных (нижний график, зеленый цвет) сообщений от времени работы компонента



Рисунок П.30 – Зависимость длины очереди от времени работы одного AMS Logs Handler при нагрузке 300,3записей/с

зависимость длины очереди сообщений от времени работы компонента (верхний график); зависимости числа полученных (нижний график, цвет желтый), обработанных (нижний график, цвет красный) и не обработанных (нижний график, зеленый цвет) сообщений от времени работы компонента

При интенсивности генерации записей 400 записей/с через 20 минут после начала теста наблюдался критический рост очереди. На рисунке П.30 этот интервал отмечен вертикальными линиями. 1 – начало роста, 2 – конец.

Таким образом, результаты выполнения сценария с учетом сформулированного критерия успешности его выполнения позволили установить, что нагрузка в 300 записей/с является допустимой для AMS Logs Handler.

В таблице П.13 приведены сводные результаты тестирования, полученные при выполнении третьего сценария.

Таблица П.13 – Сводная таблица результатов тестирования по третьему сценарию

Сценарий	Продолжительность (секунды)	Сгенерировано записей	Интенсивность генерации	Размещено в БД
Сценарий 3 (a)	60 x 60	1943334	333,3 записей/с	1941621
Сценарий 3 (b)	60 x 20	2132000	400 записей/с	2133609

*Оценка полноты синтезируемых моделей объектов и процессов в ППСМ СКЦТ.* Полнота синтезируемых моделей объектов и процессов оценивалась исходя из числа своевременно выявленных и локализованных ошибок. Проводился мониторинг сегмента крупной сети кабельного телевидения (Charter operator, USA, St-Louis segment) на интервале времени в один месяц (07/08 - 08/07/2017). Сегмент включал несколько десятков тысяч ресиверов различных типов, которые обслуживались несколькими серверами (в штатном режиме двумя), объединенными в кластер.

За исследуемый период ППСМ СКЦТ были зафиксированы проблемные ситуации на 29 ресиверах из всей популяции. Общее число выявленных ошибок в работе ТВР составило 2422. Данные по числу ТВР, на которых возникали ошибки, и типам ошибок приведены в таблице П.14.

Таблица П.14 – Статистические данные по типам ошибок, зарегистрированным при работе ТВР

Тип ошибки	ТВР, на которых она наблюдалась ошибка, ед	ТВР, на которых наблюдалась ошибка, %
Перезагрузка ТВР	25	59.52
Ошибка SDV	10	23.81
Ошибка AVN	14	33.33
Ошибка “черный экран”	16	38.1

Распределение ошибок по времени их возникновения по всем ресиверам приведено на графике (рисунок П. 31). Распределение ошибок по устройствам представлено в виде гистограммы (рисунок П. 32), а также в табличном виде (таблица П.15).

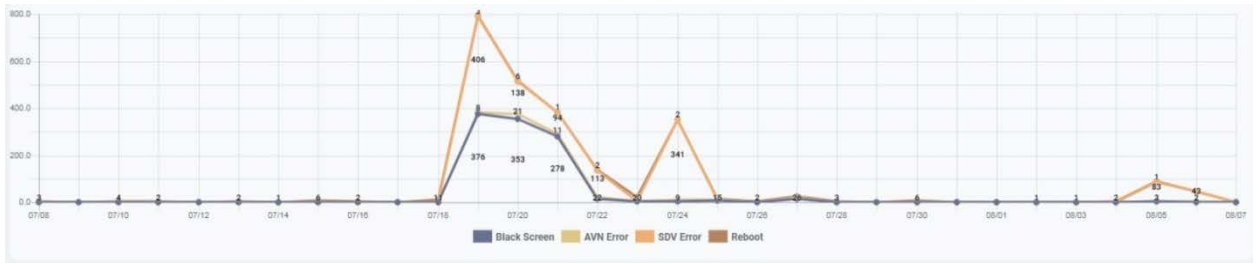


Рисунок П. 31 – Распределение типов ошибок ТВР по времени возникновения

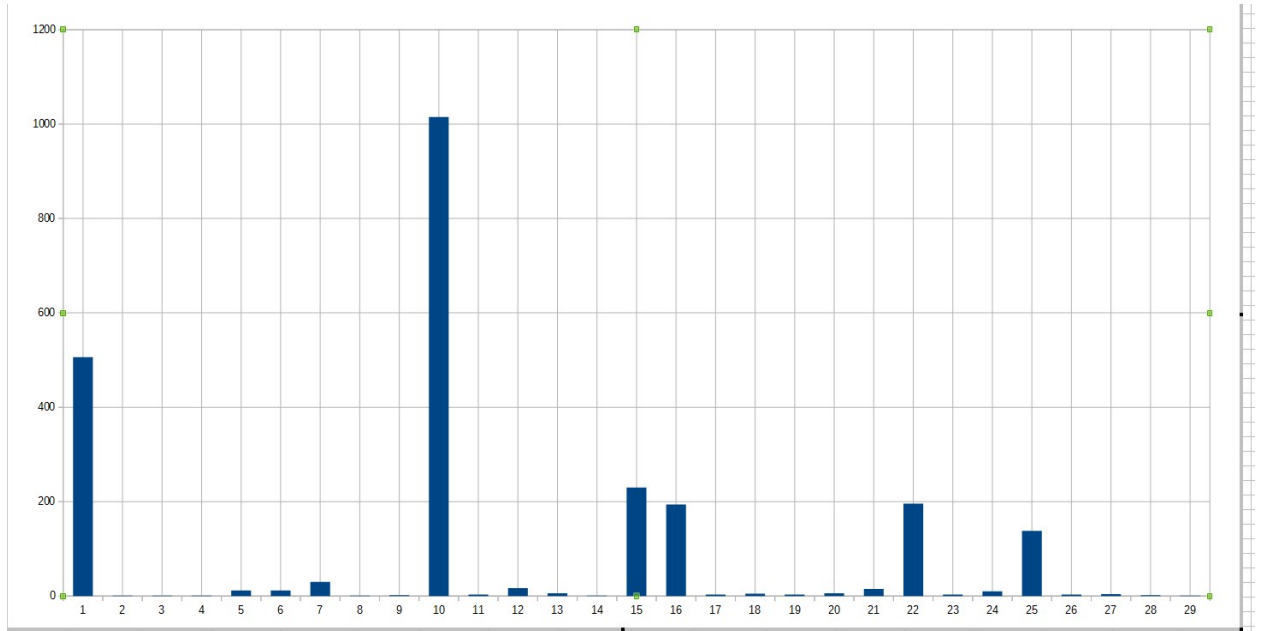


Рисунок П.32 – Распределение числа ошибок по отдельным ТВР

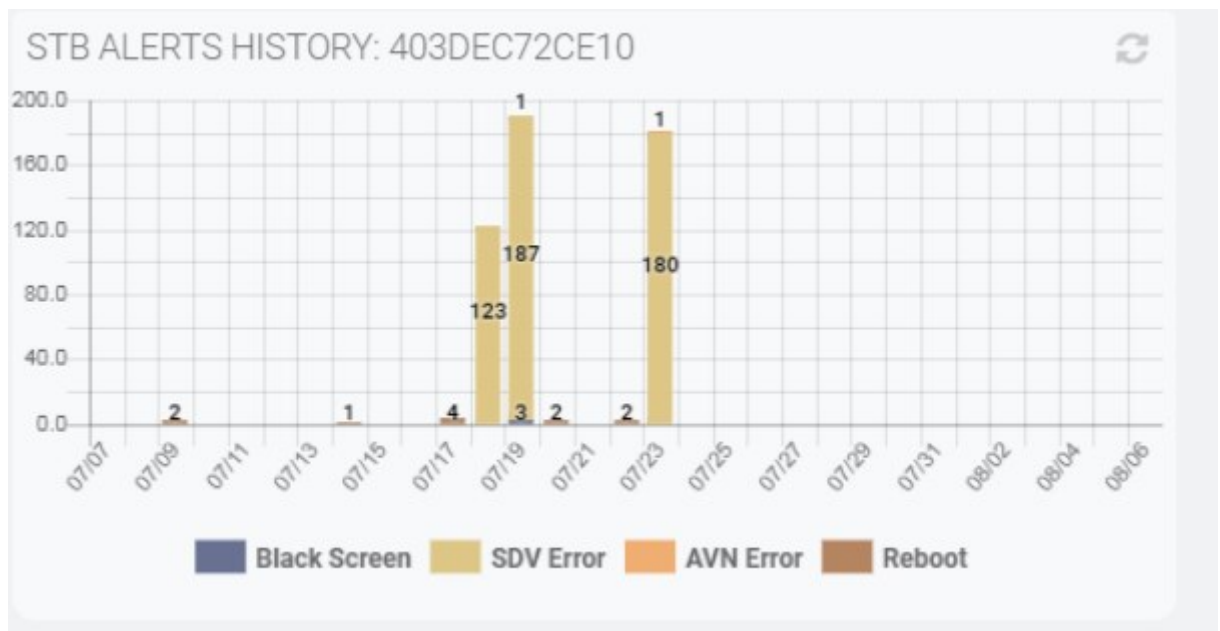
Таблица П.15 – Распределение числа ошибок по отдельным ТВР

№	Уникальный идентификатор ТВР	Число ошибок
1	403DEC72CE10	506
2	6CB56B800D40	1
3	6CB56B8056DC	1
4	6CB56BBA4D64	1
5	6CB56BBA4F0C	12
6	6CB56BBA4FA8	12
7	6CB56BBA5104	30
8	6CB56BBA51B8	1
9	6CB56BBA6128	2
10	6CB56BBA61D8	1015
11	6CB56BBA62EC	3
12	6CB56BBA6534	17
13	CC4EEC637120	6
14	CC4EEC65B0FC	1
15	CC4EEC65B158	230
16	CC4EEC66854C	194
17	CC4EEC668734	3
18	CC4EEC668DEC	5
19	CC4EEC66AA04	3
20	CC4EEC66AF58	6
21	CC4EEC66B388	15

№	Уникальный идентификатор ТВР	Число ошибок
22	CC4EEC66CCC4	196
23	CC4EEC66CD08	3
24	CC4EEC66CD94	10
25	CC4EEC66CF20	138
26	CC4EEC66D328	3
27	CC4EEC66D390	4
28	CC4EEC66DA34	2
29	CC4EEC66E4D8	1

Распределение типов ошибок для отдельных устройств представлено на рисунке П.

33.



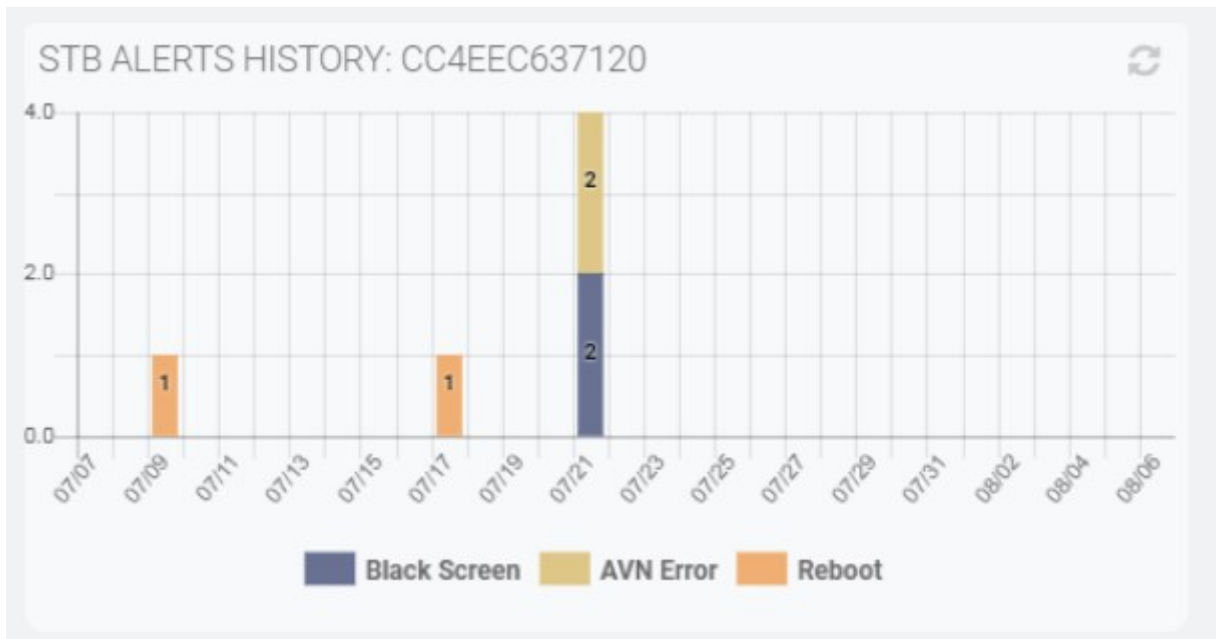
а) ошибки, зарегистрированные на ТВР 403DEC72CE10



б) Ошибки, зарегистрированные на ТВР 6CB56B800D40



в) Ошибки, зарегистрированные на ТВР 6CB56BBA4F0C



в) Ошибки, зарегистрированные на ТВР CC4EEC637120

Рисунок П.33 – Распределение типов ошибок для отдельных устройств

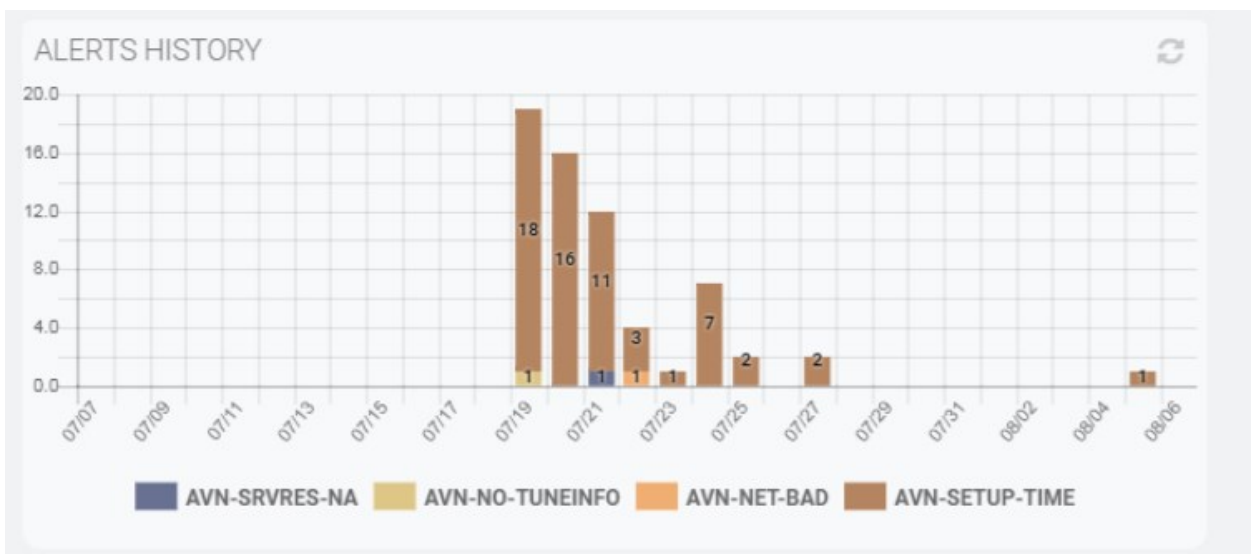
Статистические данные по выявленным причинам возникновения ошибок приведены в таблице П.16. Распределения причин возникновения различных типов ошибок во времени показаны на рисунке П. 34.

Таблица П.16 – Статистические данные по выявленным причинам возникновения ошибок

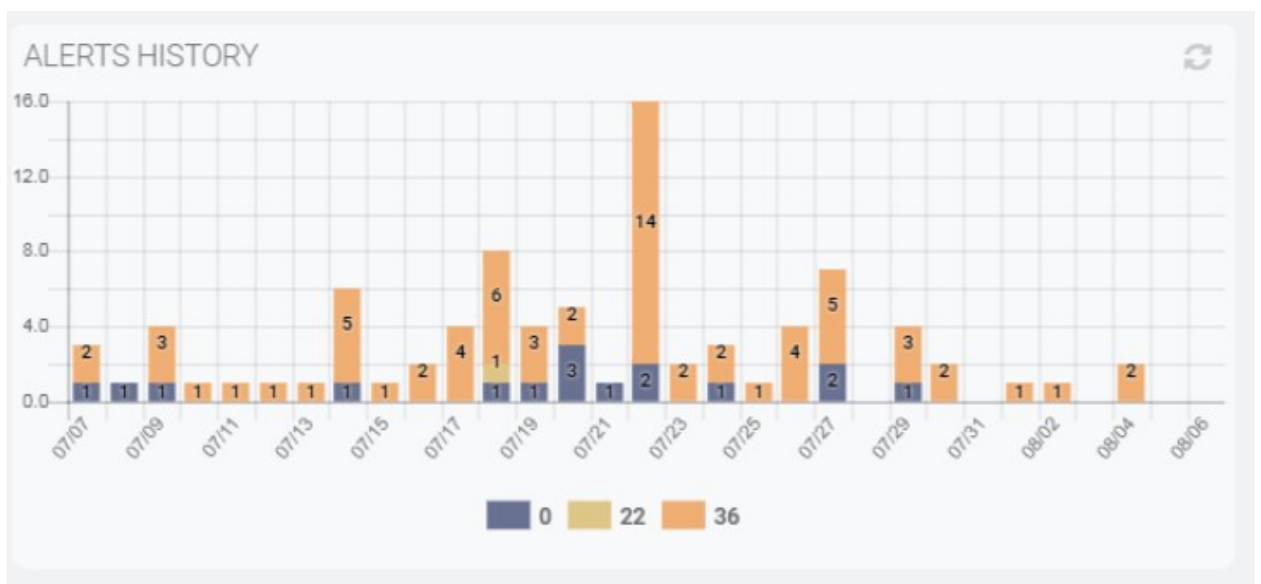
Тип ошибки	Причина	Число зарегистрированных ошибок



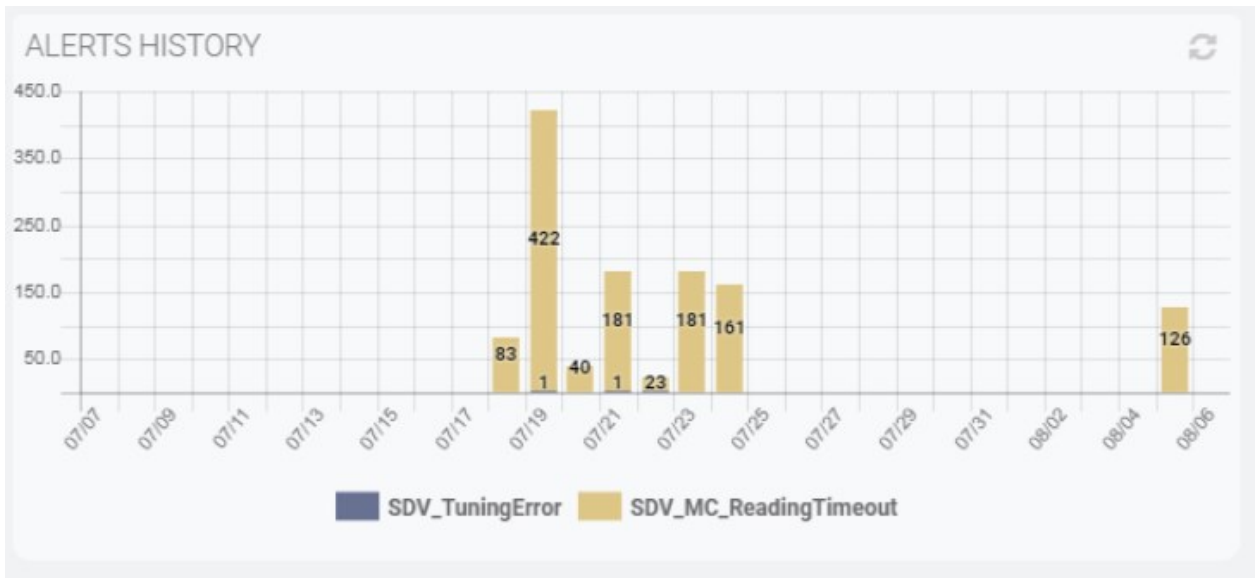
sdv_err_type	"reason": 27	3
	"reason": 28	1216
bs_err_type	"reason": 0	1025
	"reason": 1	2
	"reason": 2	26
reboot_type	"reason": 36, "subreason": 0	68
	"reason": 22, "subreason": 0	1
	"reason": 0, "subreason": 139	14
	"reason": 0, "subreason": 0	2



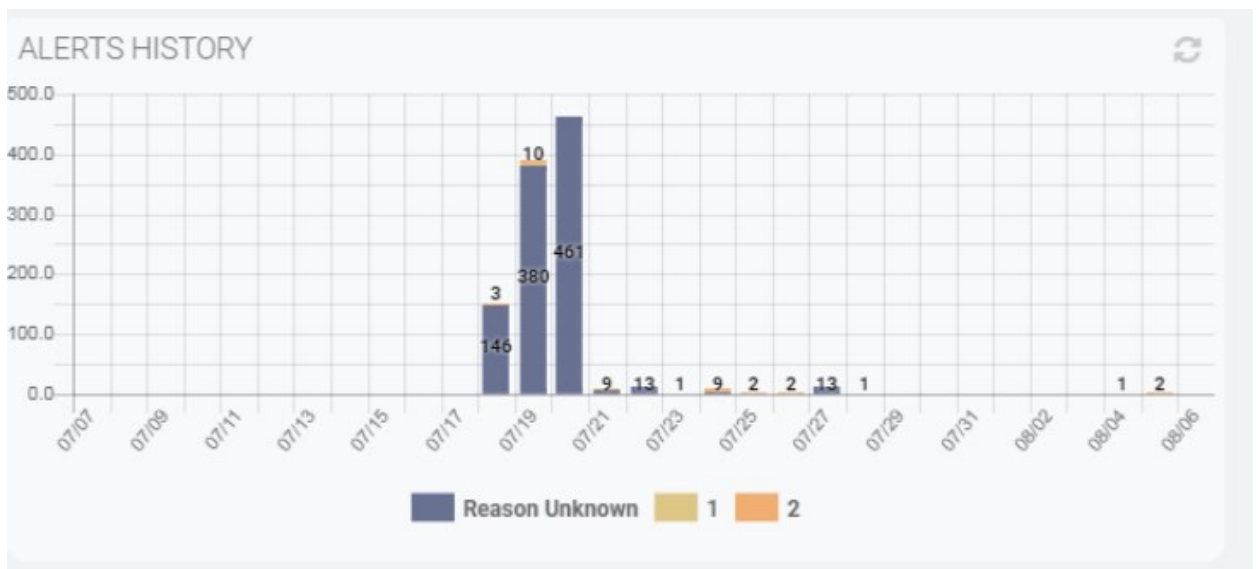
а) Распределение причин возникновения ошибок типа AVN (отсутствует AVN-канал)



б) Распределение причин возникновения ошибок типа Reboot (перезагрузка ТВР)



в) Распределение причин возникновения ошибок типа SDV (отсутствует SDV-канал)



г) Распределение причин возникновения ошибок типа “Черный экран” (отсутствует изображение)

Рисунок П.34 – Распределения причин возникновения ошибок различных типов

Для оценки полноты синтезированных моделей было проведено сравнение ошибок, зарегистрированных ППСМ СКЦТ, и ошибок, зафиксированных специалистами службы поддержки. Установлено, что системой выявлены все возникшие за рассматриваемый период ошибки.

*Таким образом, результаты тестирования позволили установить, что ППСМ СКЦТ обеспечивает полноту синтезируемых моделей ТВР для сегмента сети оператора “Charter”, развернутого в Сент-Луис.*

Серверными компонентами ППСМ СКЦТ было зарегистрировано существенно больше ошибок. На одном из тестовых периодов, в частности, с 07/24 по 08/08/2017, было

зарегистрировано 1706933 ошибок. Распределение типов ошибок приведено в таблице П.17, их распределение во времени представлено на рисунке П.35.

Таблица П.17 – Статистические данные по типам зарегистрированных ошибок серверными компонентами

Тип ошибки	Число ошибок, ед	Число ошибок, %
CID	1414484	82.86
AM	554	0.032
EPG	254071	14.8
LIN	20937	1.2
CORE	107	0.006
SET	15868	9.29
SI	0	
SMD	0	
DVRM	4	
DVR	902	0.052
PPV	0	
DSG	2	
EPGM	2	

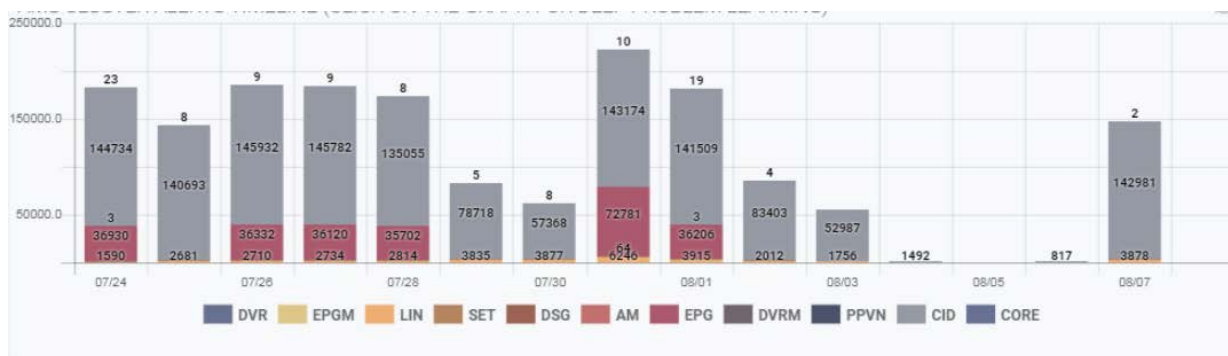


Рисунок П.35 – Распределение типов ошибок сервера по времени возникновения в период с 07/24 по 08/082017

Аналогичные данные по ошибкам сервера в период с 08/06 по 08/15 показаны на рисунке П.36.

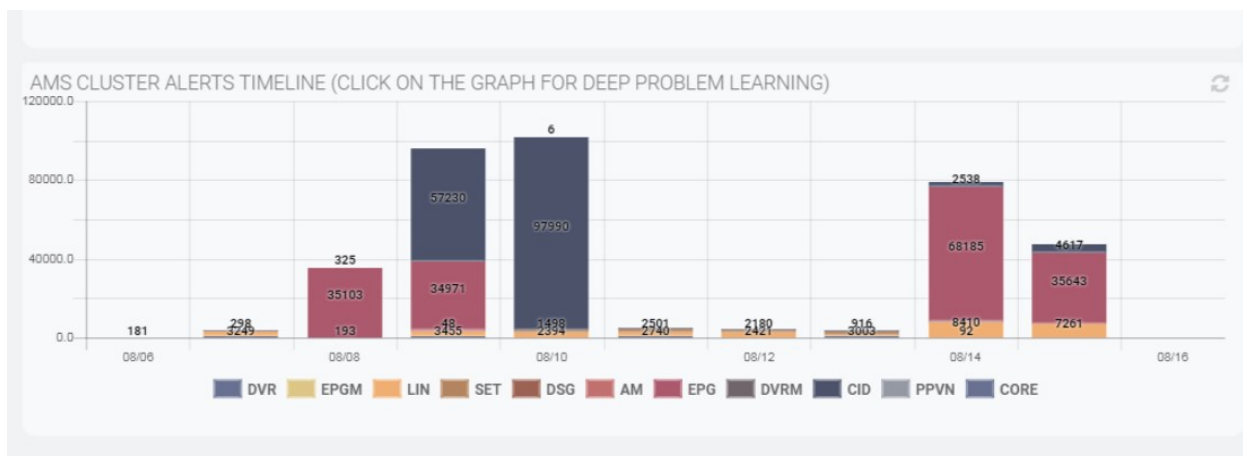


Рисунок П.36 – Распределение типов ошибок сервера по времени возникновения в период с 08/06 по 08/15

Обработка лог файлов сервера с привлечением специалистов, обладающих экспертными знаниями, показала, что все ошибки системой зафиксированы обоснованно, типы ошибок определены верно.

Таким образом, результаты тестирования позволили установить, что ППСМ СКЦТ обеспечивает полноту синтезируемых моделей сервера для сегмента сети оператора “Charter”, развернутой в Сент-Луис.

#### П.6. Рекомендации по построению моделей наблюдаемых объектов и процессов мониторинга в области телекоммуникаций

При построении моделей наблюдаемых объектов и процессов мониторинга средствами ППСМ СКЦТ предоставляется возможность настраивать процессы сбора и обработки данных. При настройке процессов сбора данных определяются следующие параметры:

- 1) временной интервал распределения (“размазывания”) отправки сообщений с ТВР; может использоваться рандомизация при отправке сообщений;
- 2) частота опроса ТВР;
- 3) уровень детализации сообщений, выводимых в лог файлы;
- 4) уровень агрегирования данных о состоянии ТВР;
- 5) уровень агрегирования данных о состоянии сервера;
- 6) способы агрегирования данных для различных групп ТВР;
- 7) состав отслеживаемых событий на ТВР, на сервере;
- 8) состав правил, запускающих выполнение действий на ТВР. Такими действиями могут являться как действия по сбору данных, так и различные управляющие воздействия;
- 9) необходимость в регулярном мониторинге, реализуемом на ТВР, способ его выполнения (конфигурация резидентного агента);

10) тип и объем используемых хранилищ данных или файлов на серверах, политика их очистки;

11) состав и способы коммуникации со сторонними компонентами и системами;

12) способы отправки управляющих команд на ТВР – широковещательный, одноадресный, многоадресный;

13) группировка ТВР. Группировка может учитывать логическую сетевую топологию, возможности аппаратного и программного обеспечения, предпочтения пользователей и другие влияющие факторы;

14) способы оценки загруженности сети ( сетевого трафика) перед отсылкой больших объемов данных с ТВР;

15) состав системных событий, на которые должны подписываться компоненты ТВР для мониторинга;

16) состав и объем регулярно отправляемых на сервер данных. Например, могут формироваться отчеты об использовании памяти ТВР, отчеты о производительности ТВР, отчеты о состоянии сети и другие.

Отчет об использовании памяти ТВР может включать: объем свободной памяти, объем свободной видеопамати, размер кеша ассетов, размер кеша программной платформы ТВР, объем свободной памяти ОС, наличие ошибок в свободной памяти ОС, наличие ошибок в свободной памяти ассетов, наличие ошибок в свободной видеопамати, наличие ошибок в свободной памяти ОС, ошибки в кеше ассетов, ошибки в программной платформе ТВР, общее количество ошибок в памяти.

Отчет о производительности ТВР обычно имеет следующую структуру: среднее время переключения канала, среднее время обработки нажатия кнопок, среднее количество кадров в секунду, средняя загрузка ЦП, объем ресурсов ЦП, потребляемых программной платформой ТВР, количество ошибок относительно среднего времени переключения канала, количество ошибок относительно среднего времени обработки нажатия клавиш на пультах управления пользователями, общее количество ошибок работы.

Отчет о состоянии сети может содержать следующие данные: ошибки TCP/ IP, ошибки карусели, средняя пропускная способность сети, таймаут чтения, таймаут установки соединения, загрузка сети, ошибки использования сети, общее количество ошибок сети.

Параметры учитываются в логических правилах в соответствии с которыми осуществляется сбор данных в ППСМ СКТЦ. Ниже приведены примеры отдельных правил.

### Правила отправки сообщений о возможных ошибках с ТВР на сервер:

1. Предупреждение об ошибке того же типа отсылается через заданный (конфигурируемый) интервал времени. Может ожидаться, что на этом интервале ошибка такого же типа (конфигурируемый параметр) должна возникнуть несколько раз (конфигурируемый параметр).
2. Если ошибка того же типа была ранее отправлена на сервер, при этом интервал времени с момента отправки меньше заданного (конфигурируемый параметр), то повторная отправка может не выполняться.
3. ТВР должна получить от сервера подтверждение о доставке:  
 если подтверждение не получено, то следует ожидать следующей ошибки в течение заданного промежутка времени (конфигурируемый параметр);  
 в случае возникновения новой ошибки необходимо отослать сообщение на сервер;  
 если ошибка не возникает, то возникновение следующей ошибки ожидается в течение заданного промежутка времени (конфигурируемый параметр).

### Общие правила формирования сообщений:

1. Сообщения могут содержать дополнительную информацию: если используется технология “размазывания” при отправке, то в начале сообщения должно указываться время регистрации ошибки.
2. Сообщения могут быть с подтверждением и без подтверждения:  
 если сообщение передает некоторую информацию, например, информацию о перезагрузке ТВР, (конфигурируемый параметр), то должно формироваться сообщение с подтверждением,  
 если информация носит уведомительный характер (конфигурируемый параметр), то при ее передаче не требуется подтверждения.

### Правила отправки технических сообщений с ТВР на сервер:

1. При изменении технических параметров ТВР, следует выполнить передачу новых значений на сервер до передачи других сообщений

### Правила сбора данных различных типов:

1. Телеметрические данные или их часть (конфигурируемый параметр) следует выгружать с группы ТВР (конфигурируемый параметр) с заданной частотой, например, раз в час или раз в сутки (конфигурируемый параметр) при выполнении набора условий, например, загрузки сети меньше заданной (конфигурируемый параметр).
2. Одиночные системные сообщения (heartbeats) следует передавать с заданной частотой, например, один раз в минуту (конфигурируемый параметр).

### Правила управления ТВР и ее ресурсами:

1. Если ресурсы ТВР, необходимые для выполнения запросов потребителей исчерпаны, т.е. достигнут некоторый порог (конфигурируемый параметр), то ресурсы, занятые задачами мониторинга, частично освобождаются (конфигурируемый параметр).  
 Таким образом, конфигурируется, например, допустимый объем памяти, который можно использовать для хранения диагностических данных и другие ресурсы.
2. Управляющие правила существенно зависят от моделей ТВР. Строятся семейства правил, которые могут иметь вид:  
 если <конфигурируемое условие>, то выполнить очистку памяти <параметры очистки>;  
 если <конфигурируемое условие>, то выполнить остановку сервисов <перечень сервисов>, <параметры остановки>;  
 если <конфигурируемое условие>, то выполнить перезагрузку <перечень параметров перезагрузки> и другие.

Общее число таких правил в достаточно масштабных системах может составлять несколько тысяч.

К параметрам, определяющим логику обработку данных, относятся:

- 1) распределение логики обработки между ТВР и серверной частью;
- 2) распределение задач обработки между центральным сервером и обработчиками сообщений (локальными серверами).

Все операции, связанные с обработкой ошибочных ситуаций, могут выполняться непосредственно на ТВР. Возможно все данные передавать на сервер. В первом случае, сеть передачи данных не испытывает дополнительных нагрузок, во втором случае не затрачиваются ресурсы ТВР на действия, напрямую не связанные с реализацией пользовательских функций. Существуют многие промежуточные варианты. Эти варианты предусматривают частичную обработку данных на ТВР и частичную обработку на сервере. Ниже рассматриваются несколько вариантов определения гибкой логики обработки.

Сценарии агрегирования данных могут быть определены следующим образом. События ошибок SDV (и/или DVR/ EPG и пр.) накапливаются в течение некоторого промежутка времени на ТВР. По окончании заданного временного интервала события отсылаются пакетом на сервер (пакетная передача данных требует меньше ресурсов). При превышении заданного допустимого числа ошибок в течение этого промежутка времени, на сервер отправляется срочное сообщение со специальной пометкой (чтобы в случае необходимости оно могло быть выведено на экран оператора). В таком варианте агрегирования на стороне ТВР осуществляется только сбор данных.

Возможен мониторинг состояния ТВР на основе контроля значений параметров на самом устройстве. Логика обработки параметров на ТВР следующая. ТВР с заданной периодичностью проверяет текущие значения некоторого набора параметров производительности, например, среднее время переключения канала, среднее время обработки нажатия кнопок, средний уровень загрузки ЦП, общее количество ошибок работы. При достижении пороговых значений за заданный период ТВР освобождает память и отправляет соответствующее уведомление на сервер. При таком варианте непосредственно на ТВР, кроме агрегирования, так же осуществляется часть обработки.

Отдельный сценарий обработки данных определяется для случаев, если на ТВР произошла серьезная ошибка, которая привела к перезагрузке устройства. В таких случаях ТВР немедленно после завершения загрузки может отправлять на сервер соответствующее уведомление. На сервер, кроме уведомлений, могут поступать также крэш-дампы с ТВР.

На сервере полученные уведомления регистрируются и выполняется обработка крэш-дампов. Возможна более тонкая настройка работы с данными о перезагрузках – например, если настройка ТВР предусматривает отправку крэш-дампов, то уведомление о перезагрузке может не отправляться по сети. Сервер может регистрировать перезагрузку в момент получения дампа. За счет этого, в случае массовых перезагрузок ТВР, нагрузка на сеть может быть несколько снижена. Такой способ применим только в случаях, когда не требуются отправки уведомлений в реальном времени. Или, наоборот, правила могут предусматривать отправку с определенных групп ТВР вместе с уведомлением о перезагрузке также последнего крэш-дампа с уменьшенной относительно базовой конфигурации задержкой для оперативной обработки.

Во многих случаях одинаковые ошибки возникают не на отдельных ТВР, а сразу на многих устройствах. Если некоторая ошибка наблюдалась на небольшом числе устройств, то информация о такой ошибке может быть получена и обработана от каждого из устройств в отдельности. Однако, если ошибка носит массовый характер, то требуется установить общие параметры для группы ТВР, на которых зафиксированы одинаковые ошибки. Исходя из выявленных группирующих параметров, определяются дальнейшие способы локализации ошибок. Группирующие параметры могут относиться как к параметрам ТВР, так и параметрам серверов их обслуживающих, сети передачи данных и другим. Группировка по параметрам ТВР выполняется локальными серверами, по другим параметрам - центральным сервером. Информация о группирующих параметрах позволяет сократить объем данных, которые необходимо собрать об ошибочной ситуации для ее локализации и устранения, сокращает время восстановления работоспособности сети.

Сценарии, связанные с получением и обработкой лог файлов, как правило, реализуются на центральном сервере. Лог файлы могут передаваться с ТВР на сервер, после чего они размещаются в специальном хранилище. Нет технической возможности получать и обрабатывать лог файлы в непрерывном режиме со всех ТВР. Кроме того, регулярный сбор таких данных, как правило, оказывается крайне неэффективным. Необходимость в данных, содержащихся в лог файлах, возникает при локализации ошибочных ситуаций. При этом сервером инициируется получение лог файлов с ТВР и выполняется их обработка. Также обработка лог файлов может потребоваться инженерам службы поддержки для решения задач по сопровождению СКЦТ, в том числе, оказания удаленной помощи пользователям конкретных устройств.

В целом, результаты эксплуатации систем показали, что настройка процессов сбора и обработки данных в соответствии с рекомендациями позволяет существенно улучшить возможности ППСМ СКЦТ.



## Приложение 2. Акты внедрения

ООО «Девелопонбокс»  
Общество с ограниченной  
ответственностью  
«Девелопонбокс»

Zodiac Systems Inc

108 B New South Road Hicksville, NY 11801

Tel +1 516 619 3170, e-mail: info@zodiacsystems.com



### АКТ

О реализации научных результатов Жуковой Наталии Александровны  
в российско-американской компании Zodiac Systems Inc

Настоящим актом подтверждается, что при разработке новых систем оперативного мониторинга и управления телекоммуникационными сетями операторов кабельного телевидения, в том числе, крупных операторов кабельного телевидения США и Канады, используются следующие научные результаты диссертационного исследования Жуковой Н.А:

- многоуровневые модели объектов мониторинга и методы их построения, включая методы многоуровневого синтеза моделей процессов и программ мониторинга;
- методы и модели многоуровневой адаптивной трансформации данных мониторинга;
- методы и модели разработки проблемно- и предметно- ориентированных систем построения моделей объектов по данным мониторинга.

Результаты, полученные в ходе применения новых систем мониторинга и управления, показали, что они позволяют оперативно выявлять и устранять 98,5% ошибок в работе систем кабельного телевидения, а также снизить стоимости проектирования и сопровождения систем мониторинга и управления, применяемых операторами кабельного телевидения.

Акт составлен в трех экземплярах.

Генеральный директор  
российского представительства  
Zodiac Systems Inc



Шипкова А.В.

07 октября 2019



Акционерное общество  
**«Научно-инженерный центр  
 Санкт-Петербургского электротехнического университета»**

194021, Санкт-Петербург, ул. Политехническая, д. 22, лит. К, пом. 289  
 Почтовый адрес: 194021, Санкт-Петербург, ул. Политехническая, д. 22, лит. Н  
 ОКПО 72491481, ОГРН 1047855001270, ИНН/КПП 7813300797/780201001  
 тел.: (812) 703-75-84 e-mail: info@nicetu.spb.ru  
 тел./факс: (812) 703-75-83 http://www.nicetu.spb.ru

УТВЕРЖДАЮ

Генеральный директор  
 АО «НИЦ СПб ЭТУ»



А.А. Дыкало

2019 г.

Акт

о реализации результатов докторской диссертации  
 Жуковой Наталии Александровны в АО «НИЦ СПб ЭТУ»

Комиссия в составе:

председателя – Постникова Е.В., д.т.н. профессора, заместителя генерального директора - главного конструктора АО «НИЦ СПб ЭТУ»; членов комиссии:

Вайнтрауба А.И., к.в.н. доцента, заместителя главного конструктора УГК;

Васильева А.В., к.т.н. доцента, начальника отдела системного анализа измерительной информации ДПО – установила, что разработанные Жуковой Н.А. автоматные модели объектов мониторинга и методы их многоуровневого автоматического синтеза, методы многоуровневого автоматического синтеза автоматных моделей процессов и программ мониторинга, методы и модели многоуровневой адаптивной трансформации данных мониторинга, а также методы и модели разработки проблемно- и предметно- ориентированных систем построения моделей объектов по данным мониторинга использованы при выполнении следующих НИР и ОКР: ОКР «Поиск-СНВ», ОКР «Адекватность», ОКР «Ленинградка», ОКР «Математика-ПИК», ОКР «Модернизация», ОКР «Перспектива-ПИК», ОКР «ИАС-М», ОКР «Признак», ОКР «Указчик-ВКО/ЛТК ОДПП и ОРИ», НИР «Листва-2000-НЦ».

Новые модели и методы многоуровневого автоматического синтеза применяются для решения задач восстановления характеристик объектов



космической техники (ОКТ) по данным, передаваемым с них во время пусков, задач анализа состояния ОКТ по результатам измерений их параметров при комплексной подготовке к пускам и их проведении и ряде других.

Применение предложенных моделей и методов автоматического синтеза и моделей многоуровневой адаптивной трансформации данных позволило существенно сократить время обработки данных и повысить точность и достоверность результатов их обработки. Время обработки, как правило, занимает секунды, не превышает нескольких минут. Обеспечивается достоверное решение 85...95% прикладных задач, точность обработки составляет порядка 98%.

Применение новых методов и моделей разработки проблемно- и предметно- ориентированных систем построения моделей объектов позволило снизить стоимость создания систем более чем в два раза, стоимость сопровождения систем стала требовать незначительных ресурсов.

Также следует отметить высокую востребованность новых программных систем. В настоящее время специалисты активно используют установленные на объектах системы построения моделей объектов мониторинга при решении 85...95 % прикладных задач.

Акт составлен в 3 (трех) экземплярах.

Председатель комиссии,  
Заместитель генерального директора -  
главный конструктор, д.т.н.



Е.В. Постников

Члены комиссии:  
Заместитель главного конструктора  
УГК, к.в.н.



А.И. Вайнтрауб

Начальник отдела системного анализа  
измерительной информации ДПО,  
к.т.н.



А.В. Васильев



Акционерное общество  
**«Научно-инженерный центр  
 Санкт-Петербургского электротехнического университета»**

194021, Санкт-Петербург, ул. Политехническая, д. 22, лит. К, пом. 289  
 Почтовый адрес: 194021, Санкт-Петербург, ул. Политехническая, д. 22, лит. Н  
 ОКПО 72491481, ОГРН 1047855001270, ИНН/КПП 7813300797/780201001  
 тел.: (812) 703-75-84 e-mail: info@nicetu.spb.ru  
 тел./факс: (812) 703-75-83 http://www.nicetu.spb.ru

УТВЕРЖДАЮ

Генеральный директор  
 АО «НИЦ СПб ЭТУ»



А.А. Дыкало  
 «20» сентября 2019 г.

Акт

о реализации результатов докторской диссертации  
 Жуковой Наталии Александровны в АО «НИЦ СПб ЭТУ»

Комиссия в составе:

председателя – Постникова Е.В., д.т.н. профессора, заместителя  
 генерального директора - главного конструктора АО «НИЦ СПб ЭТУ»;

членов комиссии:

Вайнтрауба А.И., к.в.н. доцента заместителя главного конструктора  
 УГК;

Васильева А.В., к.т.н., начальника отдела системного анализа  
 измерительной информации ДПО – установила, что теория многоуровневого  
 автоматического синтеза автоматных моделей объектов мониторинга,  
 предложенная Жуковой Н.А., применяется при решении задач анализа  
 надежности, безопасности, выявления причин и факторов, приводящих к  
 отказам, инцидентам, авариям, катастрофам, задачи планирования  
 профилактики и предотвращения происшествий на объектах ракетно-  
 космической техники (РКТ) и наземно-космической инфраструктуры (НКИ).

Предложенные Жуковой Н.А. модели и методы многоуровневого  
 синтеза реализованы при разработке прототипов программных систем,  
 проводившейся в АО «НИЦ СПб ЭТУ» в рамках выполнения НИР  
 «Радиент», ОКР «Радиент», ОКР «Диагностика- НИЦ».

Реализация предложенных Жуковой Н.А. моделей и методов  
 многоуровневого синтеза позволила учитывать зависимости между  
 различными факторами, влияющими на состояние объектов мониторинга, а

также выявлять дополнительные влияющие факторы, которые ранее не учитывались.

Кроме того, за счет использования новых моделей и методов многоуровневого синтеза обеспечено оперативное выявление причин и факторов, приводящих к отказам, инцидентам и авариям. При этом восстанавливаемые причинно-следственные связи отражают как частные, так и системные зависимости в изменении состояний объектов мониторинга.

В целом, применение разработанных Жуковой Н.А. моделей и методов многоуровневого синтеза обеспечили возможность качественно повысить эффективность мониторинга, проводимого в интересах анализа надежности и безопасности объектов РКТ и НКИ, а также повысить эффективность проводимых мероприятий по своевременному устранению причин, приводящих к отказам, инцидентам и авариям.

Акт составлен в 3 (трех) экземплярах.

Председатель комиссии,  
Заместитель генерального директора -  
главный конструктор, д.т.н.



Е.В. Постников

Члены комиссии:  
Заместитель главного конструктора  
УГК, к.в.н.



А.И. Вайнтрауб

Начальник отдела системного анализа  
измерительной информации ДПО,  
к.т.н.



А.В. Васильев



МИНОБРНАУКИ РОССИИ  
**Федеральное государственное  
 бюджетное учреждение науки  
 Санкт-Петербургский институт  
 информатики и автоматизации  
 Российской академии наук  
 (СПИИРАН)**

14 линия, 39, Санкт-Петербург, 199178  
 Телефон: (812) 328-33-11, факс: (812) 328-44-50  
 E-mail: spiiiran@iias.spb.su,  
 http://www.spiiiran.nw.ru  
 ОКПО 04683303, ОГРН 1027800514411  
 ИНН/КПП 7801003920/780101001

УТВЕРЖДАЮ

Директор СПИИРАН  
 д.т.н., профессор РАН



Ронжин А.Л.



23 октября 2019 г.

**Акт о внедрении результатов диссертационного исследования  
 Жуковой Наталии Александровны "Многоуровневый синтез автоматных моделей  
 объектов мониторинга", представленного на соискание ученой степени доктора  
 технических наук по научной специальности 05.13.01 - системный анализ, управление и  
 обработка информации (технические системы)**

Комиссия в составе: председателя – заведующего лабораторией информационных технологий на транспорте, доктора технических наук, профессора Искандерова Юрия Марсовича, членов комиссии: ведущего научного сотрудника лаборатории информационных технологий на транспорте Ивакина Яна Альбертовича, старшего научного сотрудника лаборатории информационных технологий в системном анализе и моделировании Потапычева Сергея Николаевича, составила настоящий акт, в том, что результаты диссертационного исследования Жуковой Наталии Александровны «Многоуровневый синтез автоматных моделей объектов мониторинга» были внедрены при выполнении научно-исследовательских и опытно-конструкторских работ в лаборатории "Объектно-ориентированных геоинформационных систем" (ОКР «Алеврит», ОКР «Автоматизм», «СППР «Автоматизм», ОКР « Галтель - Алеврит», НИР ONR-Global# 62909-12-1-7013 «Decision Making Support System for Arctic Exploration, Monitoring and Governance»). С применением предложенных Жуковой Н.А. моделей и методов решались общая и частные задачи предоставления объективных данных о состоянии водной среды, включая;

- задачу оценки состояния водной среды по данным различных источников в интересах создания новых систем освещения обстановки, построенных на основе интеллектуальных геоинформационных систем;
- задачу исследования состояния водной среды Арктического региона, в частности, состояния среды Баренцева моря;
- задачу оценки состояния водной среды по данным, получаемых от специализированных средств, в частности, автономных обитаемых подводных аппаратов для обследования морского дна ограниченной акватории.

Использование новых моделей и методов многоуровневого синтеза при решении перечисленных задач позволило получить следующий положительный эффект:

- сократить время предоставления данных о состоянии водной среды. Обеспечена возможность предоставления порядка 85% данных, поступающих от сетей океанографических станций, практически без задержек;

- повысить достоверность и точность предоставляемых данных о состоянии водной среды. Результаты обработки данных по Баренцеву морю показали, что достоверность получаемых решений была увеличена в среднем на 15-17%, точность возросла на 8-15%;

- выполнять построение динамических моделей водной среды для отдельных районов.

Акт составлен в трех экземплярах.

Заведующий лаборатории

информационных технологий на транспорте

д.т.н., профессор

Искандеров Юрий Марсович

Ведущий научный сотрудник лаборатории

информационных технологий на транспорте

д.т.н., профессор

Ивакин Ян Альбертович

Старший научный сотрудник лаборатории

информационных технологий в системном

анализе и моделировании

к.т.н., профессор

Потапычев Сергей Николаевич



**УНИВЕРСИТЕТ ИТМО**

**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО  
ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ**

**федеральное государственное автономное  
образовательное учреждение высшего  
образования «Национальный  
исследовательский университет ИТМО»**

**(Университет ИТМО)**

E-mail: [spiiiran@iias.spb.su](mailto:spiiiran@iias.spb.su) Кронверкский пр-т, д. 49,

Санкт-Петербург, Россия, 197101

Тел.: (812) 232-97-04 | Факс: (812) 232-23-07

[od@itmo.ru](mailto:od@itmo.ru) | [itmo.ru](http://itmo.ru)

<http://www.spiiiras.nw.ru>

ОКПО 04683303, ОГРН 1027800514411

ИНН/КПП 7801003920/780101001

Декан факультета  
программной инженерии и  
компьютерной техники

П.В. Кустарев

« 01 » октября 2019 г.



## Акт

о реализации результатов докторской диссертации

Жуковой Наталии Александровны

в учебном процессе в Университете ИТМО

Настоящим актом удостоверяется, что результаты диссертационной работы Жуковой Н.А., посвященной разработке моделей, методов и средств многоуровневого синтеза автоматных моделей объектов мониторинга, использовались при приведении исследований, направленных на разработку перспективных медицинских информационных систем (МИС) для Национального Медицинского Исследовательского Центра им. В. А. Алмазова (ФГБУ «НМИЦ им. В. А. Алмазова») Минздрава России. Разработка новых МИС проводилась на базе Университета ИТМО с привлечением ведущих специалистов международной лаборатории «Системы поддержки принятия решений в медицине».

В ходе проведенных научных исследований применялись предложенные Жуковой Н.А. иерархические автоматные модели, а также методы и модели, обеспечивающие возможность их многоуровневого синтеза. В состав



индуктивного и дедуктивного синтеза автоматных моделей объектов мониторинга, методы многоуровневого автоматического синтеза автоматных моделей процессов и программ мониторинга, методы и модели многоуровневой адаптивной трансформации данных мониторинга. При разработке прототипов МИС применялись методы и модели разработки проблемно- и предметно- ориентированных систем построения моделей объектов по данным мониторинга.

Обработка медицинских данных проводилась с использованием разработанных Жуковой Н.А. методик построения моделей объектов по данным мониторинга для решения прикладных задач.

Применение моделей и методов многоуровневого автоматического синтеза автоматных моделей, разработанных Жуковой Н.А., в области практической и исследовательской медицины позволило значительно увеличить число решаемых прикладных задач в результате модификации постановок существующих задач, а также рассмотрения принципиально новых, ранее не решавшихся задач, требующих применения методов системного анализа.

Результаты обработки данных о кардиологических больных, полученные с применением предложенных моделей, методов и средств, показали, что при решении задач комплексной оценки состояния пациентов в 99% случаев получаемые показатели адекватно отражают динамику изменения состояния пациентов.

Также применение новых моделей, методов и средств позволило повысить результативность назначения диагностических процедур. Достоверность построенных моделей, характеризующих состояния пациентов с различными диагнозами, на основе данных ЭХОКГ исследований достигает 98%.

Кроме того, в результате обработки массивов данных, накопленных в базах данных существующей в ФГБУ «НМИЦ им. В. А. Алмазова»

медицинской информационной системы, был выявлен целый ряд зависимостей, отражающих взаимосвязи параметров основных систем организма.

Руководитель лаборатории  
«Системы поддержки принятия  
решений в медицине»,  
д.м.н., проф



Конради Александра Олеговна

Сотрудник лаборатории  
«Системы поддержки принятия  
решений в медицине»,  
д.м.н.



Лушнов Михаил Степанович



Утверждаю  
 Директор Института медицинского  
 образования ФГБУ «НМИЦ им.  
 В.А. Алмазова» Минздрава России  
 Е.В.Пармон  
 « 22 » октября 2019 г.

#### Акт

о внедрении результатов докторской диссертации  
 Жуковой Наталии Александровны в учебном процессе  
 Института медицинского образования ФГБУ «НМИЦ им.  
 В. А. Алмазова» Минздрава России

Комиссия в составе председателя комиссии – заведующий кафедрой математики и естественнонаучных дисциплин кандидат химических наук доцент Михайлова Н.В., членов комиссии: профессор кафедры математики и естественно-научных дисциплин доктор медицинских наук Лушнов М.С., доцент кафедры математики и естественно-научных дисциплин кандидат химических наук Фаткин А.Ю. составили настоящий акт о том, что результаты докторской диссертации Жуковой Н.А. "Многоуровневый синтез автоматных моделей объектов мониторинга", а именно методы многоуровневого автоматического синтеза автоматных моделей объектов и процессов мониторинга используются в учебном процессе лечебного факультета в Институте медицинского образования ФГБУ «НМИЦ им. В. А. Алмазова» Минздрава России с 2019 г. при изучении дисциплины "Медицинские информационные технологии, основы библиотечного дела".

Предложенные модели и методы применяются при изложении лекционного материала и практическом занятии по дисциплине «Медицинские информационные технологии, основы библиотечного дела»:

- в лекции №3. «Компьютерный анализ медицинских данных с использованием методов математической статистики»,
- в лекции №6. «Информационные технологии в доказательной медицине»;
- на практическом занятии №6 «Информационные технологии в доказательной медицине, основы библиотечного дела», раздел «Аналитика в доказательной медицине, основные виды многомерного математико-статистического анализа, системный анализ и синергетика в доказательной медицине».

Председатель комиссии:  
 заведующий кафедрой математики  
 и естественнонаучных дисциплин, к.х.н.

Михайлова Н.В.

Члены комиссии:  
 профессор кафедры математики  
 и естественно-научных дисциплин, д.м.н.

Лушнов М.С.

доцент кафедры математики  
 и естественно-научных дисциплин, к.х.н.

Фаткин А.Ю.

*Согласовано*

*Декан мед. факультета*  
*Кухаренко Г.А.*



**СПбГЭТУ «ЛЭТИ»**  
ПЕРВЫЙ ЭЛЕКТРОТЕХНИЧЕСКИЙ

МИНОБРНАУКИ РОССИИ  
федеральное государственное автономное образовательное учреждение высшего образования  
«Санкт-Петербургский государственный электротехнический университет  
«ЛЭТИ» им. В.И. Ульянова (Ленина)»  
(СПбГЭТУ «ЛЭТИ»)

ул. Профессора Попова, д.5, Санкт-Петербург, 197376  
Телефон: (812) 234-46-51; факс: (812) 346-27-58; e-mail: [info@etu.ru](mailto:info@etu.ru); <https://etu.ru>  
ОКПО 02068539; ОГРН 1027806875381; ИНН/КПП 7813045402/781301001

УТВЕРЖДАЮ

Директор департамента образования

*(Подпись)* М.С. Куприянов

15 октября 2019 г.



### АКТ ВНЕДРЕНИЯ

результатов диссертационной работы Жуковой Наталии Александровны  
"МНОГОУРОВНЕВЫЙ СИНТЕЗ АВТОМАТНЫХ МОДЕЛЕЙ ОБЪЕКТОВ  
МОНИТОРИНГА" в учебный процесс

Настоящим актом внедрения подтверждается, что результаты диссертационной работы Жуковой Наталии Александровны внедрены в учебный процесс Санкт-Петербургского государственного электротехнического университета «ЛЭТИ» им. В.И.Ульянова (СПбГЭТУ «ЛЭТИ»).

Новые модели и методы используются при изучении дисциплины "Теория принятия решений" в разделе "Использование наблюдений в принятии решений". Изучение дисциплины предусматривается рабочими программами подготовки бакалавров и магистрантов по направлениям "Программная инженерия" 231000.62 и "Прикладная математика и информатика" 010400.62.

Предложенные модели и методы многоуровневого синтеза рассматриваются в рамках лекционных занятий как эффективный математический аппарат, обеспечивающий возможность принятия решений о состоянии объектов различной природы на основе результатов обработки поступающих от них данных.

Новые методы и модели многоуровневой адаптивной трансформации данных используются при проведении практических занятий. Их применение позволяет выполнять обработку исходных данных, необходимую для последующего построения моделей поддержки принятия решений.

Применение новых моделей и методов разработки проблемно- и предметно-ориентированных систем построения моделей объектов позволяет разрабатывать



достаточно сложные системы, применимые для решения практических задач, в рамках выполнения лабораторных работ.

Разработанные методики построения моделей объектов по данным мониторинга для решения прикладных задач позволяют студентам успешно проводить самостоятельную работу, предусмотренную рабочей программой курса.

Акт составлен в трех экземплярах.

Зам. декана факультета компьютерных  
технологий и информатики по  
учебно-методической работе  
д.т.н.

Щеголева Надежда Львовна

15 октября 2019 г.

**УНИВЕРСИТЕТ ИТМО**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ  
РОССИЙСКОЙ ФЕДЕРАЦИИфедеральное государственное автономное  
образовательное учреждение высшего образования  
«Национальный исследовательский университет ИТМО»  
(Университет ИТМО)Кронверкский пр-т, д. 49,  
Санкт-Петербург, Россия, 197101  
Тел.: (812) 232-97-04 | Факс: (812) 232-23-07  
od@itmo.ru | itmo.ru

№ \_\_\_\_\_

**Акт****о реализации результатов докторской диссертации  
Жуковой Наталии Александровны в учебном процессе в  
Университете ИТМО**

Настоящим актом удостоверяется, что результаты диссертационной работы Жуковой Н.А. "Многоуровневый синтез автоматных моделей объектов мониторинга" нашли применение в учебном процессе при реализации образовательных программ магистратуры в Университете ИТМО на факультете программной инженерии и компьютерной техники.

Научные и практические результаты, полученные Жуковой Н.А., использованы при разработке учебно-методических комплексов и при проведении лекционных, лабораторных и практических занятий следующим дисциплинам:

а) "Методы машинного обучения" (направление 09.04.01- Информатика и вычислительная техника, образовательная программа "Компьютерные системы и технологии"; направление 09.04.04 - Программная инженерия, образовательные программы "Системное и прикладное программное обеспечение", "Технологии промышленного программирования",

"Нейротехнологии и программная инженерия", "Веб-технологии", "Мультимедиа-технологии, дизайн и юзабилити", "Программное обеспечение радиоэлектронных систем");

б) "Интеллектуальный анализ данных" (направление 09.04.04 - Программная инженерия, образовательная программа "Системное и прикладное программное обеспечение");

в) "Неклассические логики" (направление 09.04.04 - Программная инженерия, образовательная программа "Технологии промышленного программирования");

г) "Разработка интеллектуальных систем" (направление 09.04.04 - Программная инженерия, образовательная программа "Программно-информационные системы").

Первый проректор, к.т.н.



Д.К. Козлова

Декан факультета программной инженерии и компьютерной техники, к.т.н.

П.В. Кустарев

21 октября 2019 г.

### Приложение 3. Свидетельства о регистрации программ

РОССИЙСКАЯ ФЕДЕРАЦИЯ



## СВИДЕТЕЛЬСТВО

о государственной регистрации программы для ЭВМ

**№ 2019663233**

**Программа автоматического многоуровневого синтеза  
моделей объектов мониторинга**

Правообладатель: *Федеральное государственное бюджетное  
учреждение науки Санкт-Петербургский институт  
информатики и автоматизации Российской академии наук (RU)*

Авторы: *Осинов Василий Юрьевич (RU), Жукова Наталья  
Александровна (RU), Климов Николай Васильевич (RU)*

Заявка № **2019662386**

Дата поступления **10 октября 2019 г.**

Дата государственной регистрации

в Реестре программ для ЭВМ **14 октября 2019 г.**



Руководитель Федеральной службы  
по интеллектуальной собственности

*Г.П. Ивлиев* Г.П. Ивлиев



РОССИЙСКАЯ ФЕДЕРАЦИЯ



## СВИДЕТЕЛЬСТВО

о государственной регистрации программы для ЭВМ

№ 2018664168

**Программа автоматического многоуровневого синтеза  
действий интеллектуальных машин**

Правообладатель: *Федеральное государственное бюджетное  
учреждение науки Санкт-Петербургский институт  
информатики и автоматизации Российской академии наук  
(СПИИРАН) (RU)*

Авторы: *Осипов Василий Юрьевич (RU), Жукова Наталия  
Александровна (RU), Климов Николай Васильевич (RU)*

Заявка № 2018661337

Дата поступления 16 октября 2018 г.

Дата государственной регистрации

в Реестре программ для ЭВМ 12 ноября 2018 г.



Руководитель Федеральной службы  
по интеллектуальной собственности

Г.П. Ивлиев

РОССИЙСКАЯ ФЕДЕРАЦИЯ



## СВИДЕТЕЛЬСТВО

о государственной регистрации программы для ЭВМ

№ 2018612644

**Модуль построения предикторов значений параметров систем организма**

Правообладатель: *федеральное государственное автономное образовательное учреждение высшего образования «Санкт-Петербургский национальный исследовательский университет информационных технологий, механики и оптики» (RU)*

Авторы: *Муромцев Дмитрий Ильич (RU), Жукова Наталья Александровна (RU), Лебедев Сергей Вячеславович (RU)*



Заявка № 2017664016

Дата поступления 29 декабря 2017 г.

Дата государственной регистрации

в Реестре программ для ЭВМ 21 февраля 2018 г.

Руководитель Федеральной службы  
по интеллектуальной собственности

*Г.П. Иванов* Г.П. Иванов